Il futuro di Linux: Systemd ▪ I segreti di Unity 5

LINUX PRO

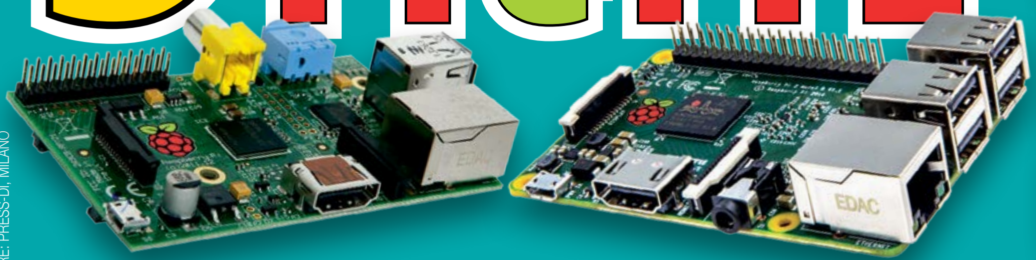
Diventa protagonista della programmazione Open Source

Non solo Linux!
Ecco i migliori sistemi operativi "cugini" del pinguino **p.48**

ACCADEMIA DEL CODICE
8 pagine di pura programmazione

Linux Pro 158 - Ottobre 2015 - Mensile - **5.90 €**

9 PROGETTI Pi e Pi 2



- Pubblica il tuo blog — Medio
- Controlla a distanza la Pi — Facile
- Crea una Smart TV — Facile
- Installa Ubuntu sulla Pi — Facile
- Crea un cabinet da sala giochi — Medio
- Progetta un allarme per la tua casa — Difficile
- Costruisci il tuo robot — Medio
- La frutta come strumento musicale col sensore capacitivo — Difficile
- Illumina la tua stanza coi LED — Medio



I contenuti del DVD

Programmi
DNSCypt 1.6.0
Guake 0.8.0
Megaglesx 3.11.1
Minilens
Plank 0.9.1
QmapShack 1.3.0
Spl 0.6.4.2
Vdrift 2012-07-22
Tor Browser 5.02

Distribuzioni
Netrunner 16.0 a 32 e 64 bit
SecurePoint 10
Debian Live 8.0.1
OpenIndiana
Haiku

Sprea Editori

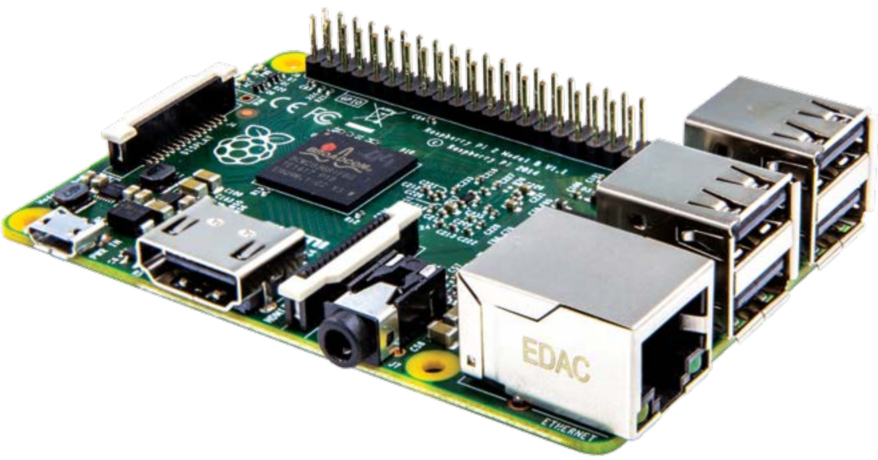
TARIFFA R.O.C. - POSTE ITALIANE SPA - ABB. POST. - DL. 353/2003 (CONV. IN L. 27/02/2004 N. 46) ART. 1, COMMA 1, S/NA - DISTRIBUTORE: PRESS-DI, MILANO

Quella voglia di creare...

C'è probabilmente un aspetto del mondo Open Source in generale che gli utenti di Windows faticano a capire. L'Open Source è sinonimo di creazione e personalizzazione e questo è qualcosa che gli utenti di sistemi chiusi come quelli di Microsoft in larga parte si perdono. E mettere mano direttamente al sistema operativo non è l'unico modo per dimostrare la passione per la personalizzazione e la creazione di funzioni e sistemi fatti esattamente come vogliamo noi. Per chi infatti vuole qualcosa di più tangibile questo mese abbiamo preparato un bellissimo articolo sulla regina dei microcomputer, la Raspberry Pi. Questa "scheda dei miracoli", piccola quanto una carta di credito o quasi e potente quanto un computer, permette infatti di creare progetti davvero strabilianti e scatena il "maker" che c'è in noi. Da mettere insieme un antifurto per la casa fino a suonare con le banane (no, non è un refuso, vi spieghiamo davvero come si fa), le 10 pagine dedicate alla Pi e alla Pi 2 vi faranno venire voglia di inventarvi sempre nuovi utilizzi per la scheda e nuovi modi per sentirvi davvero unici. Che poi è lo spirito vero dell'Open Source: completa indipendenza per permettere a tutti di avere davvero software e hardware a modo proprio!

La redazione di Linux Pro







CONTATTI

Domande alla redazione: **redazione@linuxpro.it**
Abbonamenti: **abbonamenti@linuxpro.it**
Arretrati: **arretrati@linuxpro.it**
Problemi con il DVD: **aiutocd@sprea.it**
Sito Web: **www.linuxpro.it**
Oppure inviate le vostre lettere a:
***Linux Pro*, Sprea S.p.A.,**
Via Torino 51, 20063 Cernusco S/N
Telefono: 02.92432.1

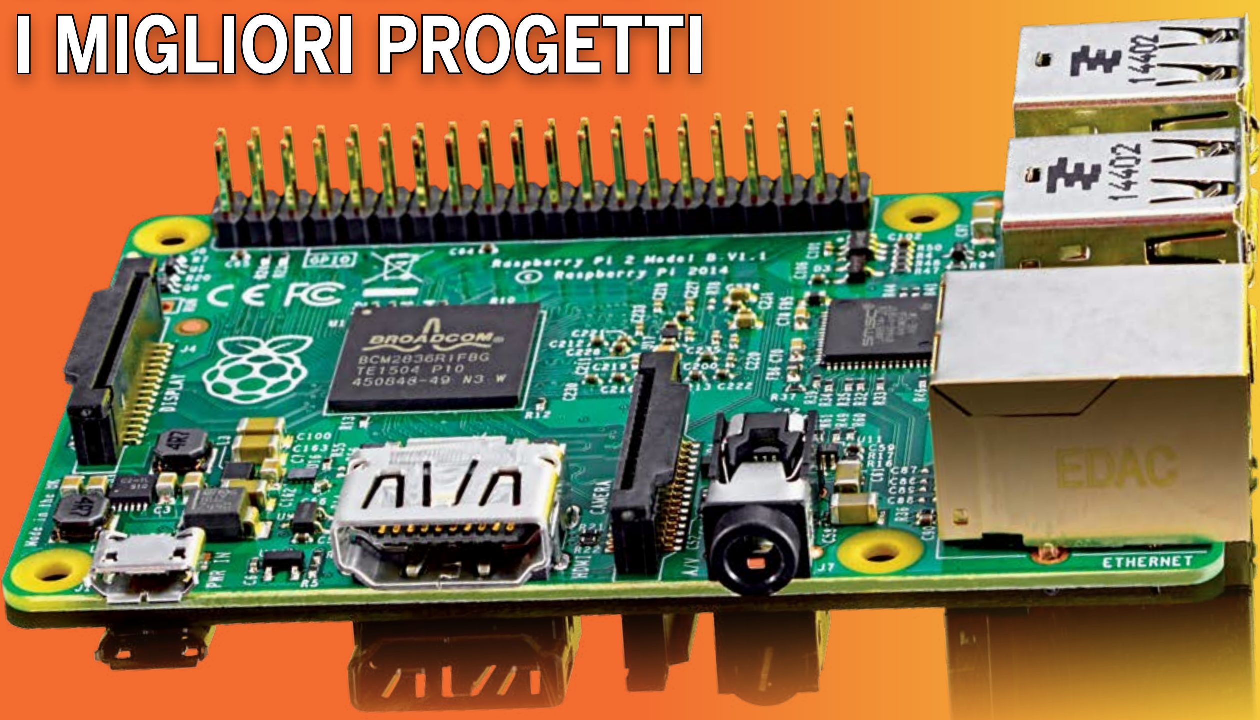
Sommario

LINUX PRO 158

Benvenuti nel centocinquantottesimo numero di Linux Pro, la guida definitiva a Linux e al mondo Open Source

In primo piano

RASPBERRY PI
I MIGLIORI PROGETTI



06 **Una guida per sfruttare al meglio Raspberry Pi in tante attività come creare una SmartTV, creare un cabinet arcade, costruire un allarme per la casa e persino suonare con la frutta!**



20 Intervista agli eroi dell'Open Source

Sommario

IL PROSSIMO NUMERO ESCE IL 12 NOVEMBRE

IL DVD IN BREVE

LATO A

DISTRIBUZIONI
- Netrunner 16



LATO B

DISTRIBUZIONI
- Debian Live 8.0.1

RIVISTA
- DNSCypt 1.6.0
- Guake 0.8.0
- Megaglesx 3.11.1
- Minilens
- OpenIndiana 151a8
- Plank 0.9.1
- QmapShack 1.3.0
- SecurePoint 10
- Spl 0.6.4.2
- Vdrift 2012-07-22
- Tor Browser 5.02

Quando trovi questo bollo negli articoli, cerca il software nel DVD

LINUX PRO dentro DVD

Newsdesk

Ogni mese tutte le novità dal mondo delle aziende e della comunità Open Source

Se hai news da segnalarci o dei commenti scrivi a newsdesk@linuxpro.it

Sistemi operativi

Ubuntu: nuove regole

Distribuire il sistema operativo è diventato ora un po' più complicato

Canonical ha cambiato la propria politica riguardo la proprietà intellettuale di Ubuntu (**http://bit.ly/LXF202-ubuntu-ip**) e ciò ha provocato una mezza rivolta, perché ha reso meno facile la distribuzione del sistema operativo e delle sue derivate. Il passo che crea più preoccupazione recita: "Qualsiasi redistribuzione di versioni modificate di Ubuntu deve essere approvata, certificata o fornita da Canonical, se si vuole associarla ai suoi Marchi. Diversamente, i Marchi vanno tolti e sostituiti. Inoltre il codice sorgente va ricompilato per creare i propri file binari".
In questo modo, Canonical vuole assicurarsi che Ubuntu non venga modificato e spacciato per una release ufficiale. Tuttavia ci sono altre implicazioni.
Nel proprio blog (**http://mjg59.dreamwidth.org/35969.html**), Matthew Garrett sostiene che "Se si applicano queste regole in senso stretto, per distribuire una versione modificata di Ubuntu, sarà necessaria l'approvazione di Canonical, oppure i marchi andranno tolti e si dovrà ricominciare da capo". Un problema sta questo passaggio: "Ubuntu è un insieme di molti prodotti, ciascuno protetto da licenza. Per stabilire ciò che si può fare con ciascuno è quindi necessario prendere atto anche delle licenze dei singoli pacchetti. I diritti concessi con quelle licenze non vengono modificati".

> "Per modificare Ubuntu, sarà d'ora in poi necessaria l'approvazione di Canonical"



› **Canonical ha compiuto un vero e proprio giro di vite riguardo la redistribuzione di versioni modificate di Ubuntu, per impedire che vengano spacciate per release ufficiali**

nella poca chiarezza di ciò che Canonical considera sua proprietà intellettuale.
Molti pacchetti, derivanti da Ubuntu, contengono la parola "Ubuntu", così come i file di aiuto. Tutto ciò ha un peso enorme sulla libertà di ridistribuire Ubuntu e, per l'ennesima volta, ha messo Canonical contro la comunità per il Software Libero. La Software Freedom Conservancy e la Free Software Foundation, che hanno collaborato per assicurarsi il rispetto della *GNU General Public License* (GPL), hanno sottolineato che, nonostante le nuove regole rispettino la GPL, lo fanno in modo insoddisfacente, perché "barano". Per esempio, c'è Il che significa che è necessario controllare ogni virgola di questa nuova politica per vedere quale contraddica la GPL.
Quindi Canonical potrà intraprendere azioni legali contro un distributore, sostenendo che la sua interpretazione è sbagliata.
La Free Software Foundation è intervenuta, sollecitando Canonical non solo a rispettare la GPL, ma anche a togliere qualsiasi restrizione ai prodotti gratuiti che distribuisce, indipendentemente dalla licenza che li protegge.
Tuttavia ha anche gettato acqua sul fuoco, aggiungendo di essere sicura della piena collaborazione da parte di Canonical. LXP

Mobile

Cinesi e smartwatch

Le vendite Android latitano ma non quelle dei produttori cinesi…

Gli unici produttori che in questo momento stanno riscuotendo un buon successo di vendite per gli smartphone sono i cinesi di Lenovo, Huawei e Xiaomi che riescono a combinare prezzo e tecnologia, anche se in qualche caso si limitano a imitare il grande assente di Berlino, quella Apple che ha appena presentato i suoi nuovi iPhone. Huawei ha mostrato in anteprima a Berlino il **Mate S**, il primo smartphone Android con display Force Touch, in grado cioè di rispondere in modo diverso a diverse pressioni sul display. Mate S si vuole contrapporre come struttura, caratteristiche e dimensioni all'**iPhone 6 Plus**: stesso schermo ma dimensioni più contenute, costruzione migliore e soprattutto prezzo dimezzato. Lenovo invece punta molto sulle batterie che rimangono il vero tallone d'Achille degli smartphone Android. Per superare il problema il gigante cinese prevede batterie da 5.000 e 4.000 mAH sui nuovi **Vibe**, una capienza quasi doppia rispetto all'attuale, con l'obiettivo di superare i due giorni di autonomia. E poi ci sono naturalmente gli smartwatch **Android Wear** che si evolvono contemporaneamente allo sviluppo del sistema operativo da parte di Google. Così Motorola ha presentato la seconda versione del **Moto 360** che questa volta si arricchisce di una versione Sport con cardiofrequenzimetro e GPS e integrato, pur conservando il prezzo di 249 euro, Asus invece ha presentato l'economico **ZenWatch 2** che sarà venduto a soli 149 euro. Si differenzia da tutti gli altri Samsung il cui **Gear S2** utilizza Tizen, ma che può vantare una versione con SIM integrata. LXP



› **Il primo smartphone Android a montare un display con tecnologia simile a quella Force Touch di Apple è il *Mate S* di Huawei**

Software Libero

Arruolato LibreOffice

Firmato l'accordo con il quale la Difesa adotterà il pacchetto per ufficio

Lo scorso 15 settembre, è stata annunciata la firma dell'accordo tra l'Associazione LibreItalia Onlus, rappresentata dal suo presidente, Sonia Montegiove, e il Ministero della Difesa, rappresentato dall'Ammiraglio di Divisione e Dirigente Generale Responsabile dei Sistemi Informativi Automatizzati, Ruggiero Di Biase, per l'adozione del pacchetto per ufficio **LibreOffice**. Le due organizzazioni collaboreranno per l'implementazione del nuovo Software Libero, sulla base del protocollo di riferimento redatto da "The Document Foundation". La Difesa si occuperà dei corsi di formazione online su LibreOffice, che verranno rilasciati alla comunità con licenza Creative Commons. LXP



RASPBERRY Pi

I migliori progetti!



Dopo l'uscita del Raspberry Pi 2, ci siamo cimentati in una serie di attività che vi aiuteranno a sfruttare al meglio questo formidabile dispositivo

Il Raspberry Pi 2 è arrivato ed è più potente di quanto ci saremmo aspettati. Per celebrare il suo avvento e le sue oltre 5 milioni di copie vendute, abbiamo deciso di realizzare una serie di progetti molto interessanti. Non importa se siete principianti o esperti. Nelle pagine che seguono troverete spunti per tutti e utili a spingere il vostro Pi oltre i suoi limiti. La maggior parte di questi possono essere realizzati anche con il Raspberry Pi Modello B. In ogni caso, abbiamo indicato le specifiche di compatibilità e gli eventuali accessori di cui avrete bisogno. I progetti su cui ci siamo concentrati abbracciano un po' tutti i campi: software, hardware e codifica. Tra le tante cose, potrete leggere una serie di nozioni che vi permetteranno di muovere i primi passi nel mondo della robotica.

> "Lasciatevi guidare dai nostri tutorial e approfondite l'interazione tra Linux e Pi"

Il Raspberry Pi è oramai diventato un vero fenomeno di massa. Sta continuando a macinare vendite e, grazie alla sua flessibilità, sempre più persone si stanno avvicinando a questo dispositivo, anche tra coloro che non hanno mai messo mano ai single-board-computer. Se quindi lo avete acquistato, oppure siete solo incuriositi, queste pagine vi forniranno una serie di spunti per utilizzarlo al meglio. Tra le altre cose, è importante ricordare come non sia fondamentale conoscere solo Pi. Gran parte dei progetti e delle utilità di Raspberry possono essere scoperte attraverso il mondo FOSS/Linux. Sarete così in grado di usare questo dispositivo anche su sistemi desktop e server.

Realizziamo un blog

Scopriamo come creare e installare un **Ghost blog** utilizzando Raspberry Pi 2 e l'eleganza degli standard **Node.js** e **Nginx**

Compatibile con tutte le Pi, ma meglio con le Pi 2

I blog non hanno decisamente bisogno di hardware potente. Grazie al nuovo Raspberry Pi 2, possiamo crearne uno estremamente reattivo sfruttando tre grandi progetti Open Source: Node.js, Nginx e la piattaforma Ghost blogging.

Tutto ha inizio con una nuova installazione di Raspbian che deve essere configurato per eseguire un server SSH. Il modo più semplice per farlo è tramite il menu **raspi-config Advanced**. Userete un client SSH per accedere al Raspberry Pi in remoto, ma prima avrete bisogno di reperire l'indirizzo IP del Pi che si ottiene tramite **ipconfig** da LXTerminal (avviatelo tramite l'icona presente sul desktop di Raspbian). Se siete connessi via Ethernet, date un'occhiata a **eth0**, se invece usate il Wi-Fi, puntate l'attenzione su **wlan0**. Cercate quindi la voce **addr inet**. Di seguito a questa, ecco che troverete l'indirizzo IP interno al Pi. Utilizzando un client SSH sul vostro computer (lo trovate in tutte le distro Linux), potrete accedere al vostro Raspberry Pi.

```
ssh pi@INDIRIZZO_IP_DEL_PI
```

Una volta connessi è necessario scaricare Node.js e passare alla sua directory, così da estrarlo e installarlo:

```
sudo wget http://nodejs.org/dist/v0.10.28/node-v0.10.28-linux-arm-pi.tar.gz
cd /usr/local
sudo tar xvzf ~/node-v0.10.5-linux-arm-pi.tar.gz --strip=1
```

Adesso scaricate e installate Ghost nella directory **home**:

```
sudo mkdir ghost
cd ghost
wget https://ghost.org/zip/ghost-0.5.8.zip
unzip https://ghost.org/zip/ghost-0.5.8.zip
```

Node.js usa lo strumento di packaging **npm** che sfrutteremo per installare Ghost come segue:

```
sudo npm install --production
```

Questa istruzione richiederà un certo tempo. Adesso lanciate:

```
sudo npm start
```

Tale comando avvierà il servizio Node.js ed eseguirà Ghost in modalità di sviluppo. In questo momento stiamo semplicemente provando la corretta configurazione del sistema. Per arrestare il server, premete **Ctrl+C.** Adesso installate Nginx:

```
sudo apt-get install nginx
```

Ora dobbiamo configurare Nginx per lavorare con Ghost. Nel LXTerminal, modificate la directory in cui i file di configurazione di Ngix risiedono ed eliminate il file di configurazione predefinito:

```
cd /etc/nginx/
sudo rm sites-enabled/default
```

A questo punto dovrete cambiare la directory **sites-available** e creare un nuovo file chiamato **Ghost** semplicemente tramite l'editor nano.

```
cd sites-available
sudo nano ghost
```

Questo file deve contenere la configurazione per connettere Ghost a Nginx, permettendo così agli utenti di accedere al blog:

```
server {
    listen 0.0.0.0:80;
    server_name ghostblog.com;
    access_log /var/log/nginx/*your-domain-name*.log;
    #root /home/pi/ghost;

    location / {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header HOST $http_host;
        proxy_set_header X-NginX-Proxy true;

        proxy_pass http://127.0.0.1:2368;
        proxy_redirect off;
    }
}
```

Salvate il file (**CTRL+O**) e uscite da nano (**CTRL+X**), quindi cambiate la directory e create un link simbolico:

```
ln -s sites-available/ghost sites-enabled/ghost
```

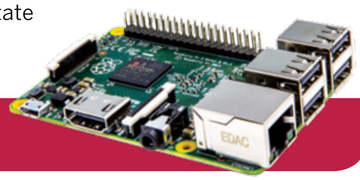
Nginx si metterà in ascolto sulla porta 90 e reindirizzerà tutto a Ghost. Infine, avrete solo bisogno di avviare Nginx e il server Ghost.

```
cd /home/pi/ghost
sudo service nginx restart
sudo npm start
```

Utilizzando un altro dispositivo, con un browser puntate all'indirizzo IP del Pi. Vedrete comparire la schermata di login di Ghost. Per saperne di più visitate **www.ghostforbeginners.com**.

Per questo progetto c'è bisogno di...

- Un Raspberry Pi 2 modello B o un Raspberry Pi 1 modello B o B+
- Raspbian OS
- Connessione Wi-Fi o Ethernet
- Server SSH che lavora sul vostro Raspberry Pi



Che cos'è LAMP?

In passato, creare un blog richiedeva un ingente quantitativo di risorse. Per cominciare, avreste dovuto installare uno stack LAMP, vale a dire Linux (sistema operativo), Apache (server Web), MySQL (database per la memorizzazione dei contenuti) e PHP (linguaggio di scripting che permette alle pagine di trasmettere informazioni a MySQL e creare contenuti HTML al volo). Questo stack viene ancora utilizzato, ma ci sono comunque progetti che si pongono lo scopo di sostituirlo. La nuova frontiera è caratterizzata da Node.js e Nginx. Il primo è un framework Javascript che viene eseguito sul server e non nel browser Web dell'utente. Questo è alla base della potenza di Ghost e sovrintende alla sua interfaccia utente e alla gestione con il database. Nginx, invece, viene utilizzato per conferire alle pagine Web la scalabilità. Si tratta di un proxy ad alte prestazioni che sta riscuotendo un discreto successo tra gli sviluppatori.

Compatibile con tutti i Pi

Controllo remoto del Pi

Grazie alla magia di **SSH** e **VNC**, impariamo a controllare a distanza il nostro Raspberry Pi

Per questo progetto c'è bisogno di...

- Un Raspberry Pi 2 modello B o un Raspberry Pi 1 modello B o B+
- Connessione Wi-Fi o Ethernet
- Tastiera, mouse e monitor per la configurazione
- Raspbian OS
- Alimentatore

Di solito, il Raspberry Pi viene utilizzato tramite tastiera, mouse e monitor. Tuttavia, questo approccio molto classico potrebbe non essere il migliore quando avete a che fare con progetti avanzati. Il controllo remoto non è certo una novità. Nel caso del Raspberry Pi, però, è un'ottima soluzione da adottare quando non siamo fisicamente nelle vicinanze del dispositivo. Nel nostro progetto installeremo un server SSH che ci permetterà di controllare a distanza il single-board-computer. Il tutto tramite il terminale di un'altra macchina. Per l'occasione, useremo anche un server VNC che consentirà di utilizzare il Raspberry in rete.

Collegate il Raspberry Pi e avviate Raspbian sul desktop. Potrebbe essere necessario effettuare il login tramite username e password. Se il Pi non si avvia direttamente, usate il comando **startx** in LXTerminal, quindi premete Invio. Se state usando un dongle Wi-Fi, assicuratevi che sia configurato e disponga di un indirizzo IP. Se invece state sfruttando una connessione Ethernet, aprendo LXTerminal, usate il comando **ipconfig** per trovare l'IP. Adesso dovete configurare il software che verrà eseguito sul PC e creare l'accesso di cui avete bisogno. Aprite LXTerminal e inserite l'istruzione che segue per impostare un server SSH:

```
sudo raspi-config
```

Spostatevi in **Advanced Option** e cercate il server SSH per attivarlo. Nel caso vi venga chiesto di riavviare il sistema, acconsentite. Con il server SSH installato, potete provare il suo funzionamento utilizzando un altro computer. Nel nostro caso, abbiamo usato un portatile con Linux Mint e SSH nel Raspberry Pi tramite il terminale:

› ***SSH*** **è incluso sia in Linux sia in Mac. Gli utenti Windows, invece, devono scaricare l'applicazione gratuita *PUTTY***

```
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.8' (ECDSA) to the list of known hosts
pi@192.168.0.8's password:
Linux raspberrypi 3.12.35+ #730 PREEMPT Fri Dec 19 18:31:24 GMT 2014 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb  3 19:17:48 2015 from 192.168.0.3
pi@raspberrypi ~ $ uname -a
Linux raspberrypi 3.12.35+ #730 PREEMPT Fri Dec 19 18:31:24 GMT 2014 armv6l
pi@raspberrypi ~ $
```



› **Per visualizzare il desktop di Raspberry Pi abbiamo usato *Vinagre* che trovate sia in Ubuntu sia in Linux Mint**

```
ssh pi@IP DEL VOSTRO Pi
```

La prima volta che ci si connette al Raspberry Pi, SSH chiederà di confermare l'effettiva autenticità del Pi tramite un'impronta digitale unica. Dopo qualche istante, dovrete reinserire la password di accesso al dispositivo. Da questo momento in poi, potrete agire con qualsiasi comando.
Con il server SSH installato correttamente, spostate la vostra attenzione sull'installazione del server VNC. Nella sessione SSH, eseguite il seguente comando:

```
sudo apt-get install tightvncserver
```

Questo consente di installare VNC. A questo punto eseguite il tipo di server:

```
tightvncserver
```

Vi verrà chiesto di immettere una password (utilizzatene una piuttosto robusta, con almeno otto caratteri). Nel terminale, vedrete quindi l'output del comando **tightvncserver** e verrete informati dell'avvio di una sessione VNC. Potrete quindi connettervi utilizzando l'indirizzo IP o il nome host seguito da :1. Adesso avrete bisogno di un client VNC da utilizzare per connettervi con il server in esecuzione sul Pi. Abbiamo utilizzato **Vinagre** che gira perfettamente su Linux Mint e Ubuntu. Apritelo, quindi fate click su **Connect**. Inserite l'indirizzo IP del vostro Raspberry Pi seguito da :1 e premete di nuovo **Connect**. Una volta inserita la password VNC, è possibile utilizzare mouse e tastiera del computer per controllare il Pi e quindi anche il desktop di Raspbian.
Per coloro che vogliono giocare a Minecraft in questo modo, al momento non è possibile farlo a causa delle modalità con cui il gioco interagisce con Raspberry Pi. In generale, comunque, questo sistema è uno dei migliori per utilizzare il dispositivo da remoto tramite un altro computer.

Cosa sono SSH e VNC?

SSH viene utilizzato dagli amministratori di sistema per lavorare in remoto con i server di tutto il mondo. Quando non potete contare su un accesso fisico al computer o al server, tramite un terminale o un'interfaccia Web è possibile utilizzare SSH.
Le connessioni di questo tipo sono cifrate ed è quindi altamente improbabile che qualcuno possa spiarle. **VNC**, invece, invia le informazioni tramite una connessione non crittografata. Questo significa che il suo uso sul Web è estremamente pericoloso, perché mette a repentaglio la sicurezza dei dati trasmessi. La soluzione migliore, in questo caso, è usare un tunneling di una connessione VNC tramite SSH. Ci sono applicazioni SSH per tutti i sistemi operativi e su Linux vengono addirittura fornite nella dotazione base del sistema. I programmi VNC, invece, sono installati in Ubuntu e Linux Mint tramite Vinagre.

Creiamo una SmartTV

Compatibile con tutti i Pi, ma meglio con Pi2

Scopriamo come realizzare un media center e rendere così il nostro TV un vero centro di intrattenimento

I modi e le forme con cui possiamo usufruire dei contenuti multimediali sono davvero molti. Tuttavia, nonostante il tempo passi, il TV rimane al centro dei nostri ecosistemi per l'intrattenimento. Quello che forse non tutti sanno è che il Raspberry Pi è sempre stato un ottimo strumento per la gestione del multimedia. Infatti, fin dai suoi albori, molti utenti lo hanno utilizzato per creare veri centri multimediali avanzati. In questo progetto, vi mostreremo come fare. Iniziamo copiando il sistema operativo in una scheda SD. Per l'occasione, abbiamo bisogno di estrarre il contenuto del file scaricato dal sito Web di OpenELEC. Per copiare l'immagine nella SD, seguite la guida proposta a questo indirizzo: **http://bit.ly/LXF_OpenELEC**. Una volta eseguita la procedura, inserite la memoria nel Raspberry Pi, quindi collegate al TV tastiera e mouse wireless, più altre connessioni come Ethernet e HDMI. Al primo avvio, OpenELEC vi condurrà attraverso una serie di schermate che permettono di configurare il media center. Una volta completata la procedura, tornerete al menu principale che è diviso in diverse sezioni. Lo scopo, infatti, è catalogare con precisione foto, video e musica. Grazie a queste tre categorie, è possibile importare facilmente una grande quantità di contenuti da svariate fonti. Per iniziare, basta collegare un disco fisso portatile USB con dei film archiviati e spostarsi nella categoria **Video**. Da qui comparirà una nuova finestra di dialogo che permette di selezionare l'unità esterna e i rispettivi file memorizzati. Naturalmente, i contenuti possono essere disponibili anche in un NAS. A tal proposito, potrete sfruttare le incredibili capacità di connessione di OpenELEC che forniscono supporto per SSH, NFS e Samba. Il media center è poi fornito di una serie di componenti aggiuntivi che permettono di migliorarne l'esperienza d'uso. Per utilizzarli, è sufficiente accedere al menu **Add-on**. Una volta trovato il plug-in a cui si è interessati, è possibile installarlo facendo click su **Install**. OpenELEC penserà a tutto il resto. Per controllare il media center, i dispositivi di input predefiniti sono tastiera e mouse wireless. Tuttavia ci sono altre soluzioni, tra cui **FLIRC**, **https://flirc.tv**. Si tratta di un ricevitore a infrarossi che può essere programmato per utilizzare il telecomando come controller per OpenELEC. In alternativa, se avete un tablet Android, si può sfruttare l'applicazione gratuita **Yatse** (**http://bit.ly/LXF_Media_Remote**), capace di trasformare il dispositivo mobile in un controller multimediale. Questa app è in grado di mostrare il catalogo dei file sullo schermo del tablet, consentendo quindi di sfogliare e navigare tra i contenuti in modo semplice e veloce.



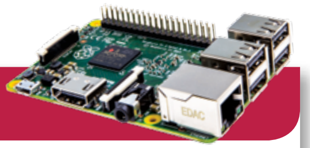
› **OpenELEC utilizza *Kodi* (precedentemente conosciuto come XBMC) ed è un media center estremamente flessibile, intuitivo e potente**



› **Le copertine degli album vengono scaricate in automatico grazie a uno strumento che scandaglia la Rete**

Per questo progetto c'è bisogno di...

- » Un Raspberry Pi 2 modello B (per risultati migliori) o un Raspberry Pi 1 modello B o B+
- » OpenELEC
- » SD/Micro SD vuota
- » Connessione Ethernet
- » Disco fisso USB o flash drive
- » Tastiera e mouse wireless



Cos'è OpenELEC

I media center sono stati tra i primi progetti con cui il Raspberry Pi si è cimentato. Sono diventati così popolari che la Raspberry Pi Foundation ha addirittura investito nel loro sviluppo. Eben Upton stesso ha confermato il suo grande interesse per **OpenELEC**, dichiarando la propria soddisfazione per l'interazione che il media center ha dimostrato con Raspberry Pi. È importante poi ricordare che Raspberry Pi 1 e 2 condividono la stessa GPU Videocore IV. Ciò significa che possono lavorare facilmente con file video di grandi dimensioni a 1.080p. L'interfaccia utente di OpenELEC richiede comunque un certo impegno anche da parte della CPU. Infatti, per ottenere le migliori prestazioni possibili, molti utenti hanno spesso eseguito un overclock del processore. Se però avete un Raspberry Pi 2 non c'è bisogno di alcuna modifica. Infatti, grazie ai miglioramenti operati a livello di CPU e RAM, la seconda versione del single-board-computer è in grado di operare con OpenELEC senza sforzi.

Compatibile solo con Pi2

Installiamo Ubuntu 14.10

Grazie al progetto **Snappy Ubuntu**, adesso è possibile installare il sistema operativo di Canonical su Raspberry Pi 2

Per questo progetto c'è bisogno di...

- Un Raspberry Pi 2 modello B
- Scheda SD da 4 GB
- Periferiche da usare con il Raspberry Pi 2 come tastiera e mouse wireless

Nel 2001, quando il Raspberry Pi è stato annunciato, in molti hanno sperato di potervi installare fin da subito Ubuntu. Tuttavia, a causa della CPU originale, la speranza non si è concretizzata. Con il rilascio del Raspberry Pi 2, però, le cose sono cambiate. Infatti, grazie alla nuova CPU Arm7, Ubuntu può finalmente essere installato sul dispositivo. In questo progetto approfondiremo montaggio e installazione della famosa distro per l'uso quotidiano.

Per prima cosa, è necessario scaricare l'immagine di Ubuntu da **http://bit.ly/Raspuntu**, quindi estrarne il contenuto tramite un gestore di archivi. Questo vi permetterà di avere tra le mani un file da 3 GB che dovrà poi essere scritto nella scheda SD tramite il comando **dd**. Collegate quindi mouse e tastiera wireless e un monitor tramite HDMI. Infine accendete il Raspberry Pi. In circa 30 secondi, vedrete comparire una schermata di accesso in cui dovete immettere la password **Linaro**. Dopo pochi secondi, il desktop viene caricato. Noterete come l'interfaccia predefinita non sia Unity, bensì la più leggera e versatile LXDE. In primo luogo, è quindi necessario configurare la connessione a Internet.

Se disponete di un dongle Wi-Fi, assicuratevi che sia inserito nel Pi, quindi accedete alle impostazioni per la connessione e selezionate **wpa_gui**. Se il dongle è elencato nel menu **Adapter**, allora è possibile connettersi tramite wireless.

In alternativa, potete usare la connessione via cavo Ethernet.



› ***Gimp* è un software che non può mancare su Ubuntu per Raspberry**



› ***Inkscape* è un'ottima risorsa da sfruttare per l'elaborazione della grafica vettoriale**

Per coloro che invece vogliono cimentarsi in un hack del Wi-Fi, è possibile creare un file di configurazione scrivendo:

```
sudo leafpad  /etc/wpa_supplicant/wpa_supplicant.conf
```

All'interno del file, inserite le seguenti istruzioni (aggiungendo il vostro SSID e la relativa password nelle rispettive sezioni):

```
network={
ssid="il SSID della vostra rete"
psk="la password"
proto=RSN
key_mgmt=WPA-PSK
pairwise=CCMP
auth_alg=OPEN
}
```

Una volta fatto, dovrete salvare e riavviare il Raspberry Pi, quindi accedere di nuovo al sistema. Adesso potrete procedere all'installazione di nuovi software. In prima battuta, dovrete assicurarvi che il sistema sia aggiornato. Aprite un terminale e inserite le seguenti istruzioni:

```
sudo apt-get update
sudo apt-get upgrade
```

Adesso installate **LibreOffice** con:

```
sudo apt-get install libreoffice
```

Questa procedura richiederà alcuni minuti, al termine dei quali avrete scaricato e installato la nota suite per l'ufficio. Altri software da installare sono **Gimp** e **Inkscape** che si rivelano molto utili per la gestione della grafica:

```
sudo apt-get install gimp
sudo apt-get install inkscape
```

Ubuntu non sostituirà Raspbian come distro predefinita per Raspberry, ma è comunque possibile utilizzarla come alternativa. Per gli ultimi sviluppi, tenete d'occhio l'indirizzo **http://bit.ly/UbuntuForRP2Forum**.

Debian root

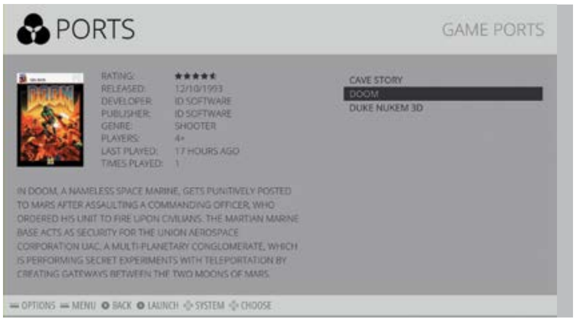
Ubuntu e Raspbian hanno in comune lo stesso sorgente originale di Debian. È quindi possibile installare le applicazioni nello stesso modo. Nel momento in cui scriviamo, si possono usare solo i repo per Ubuntu 14.10 Utopic Unicorn, ma la comunità sta lavorando per il porting dei repository di Raspbian su Ubuntu. Durante questo tutorial, abbiamo provato a vedere se fosse possibile aggiungere i repo di Raspbian alla nostra lista di fonti. Seppure nell'importazione non si sia verificato alcun problema, le più grosse difficoltà le abbiamo trovate quando ci siamo cimentati nell'installazione. Non sono infatti mancati diversi crash di sistema. Attualmente, il progetto Ubuntu per Raspberry Pi è basato su **Snappy Ubuntu**, una versione estremamente leggera del noto sistema operativo destinata a lavorare con IoT (Internet delle cose). È possibile scaricare Snappy direttamente dal sito ufficiale di Raspberry Pi.

Creiamo un cabinet arcade

Compatibilità
Tutti i Pi, ma meglio Pi2

Abbiamo nostalgia dei vecchi giochi? Ecco come realizzare una piattaforma che li riporta in vita

Tra gli anni ‘80 e ‘90 ci fu una vera esplosione di console di gioco. Chi ha qualche anno sulle spalle ricorderà sicuramente lo ZX Spectrum o il Sega Megadrive. Queste macchine sono oramai diventate dei pezzi da museo, ma ci sono ancora molti appassionati che ne sentono la nostalgia. In questo progetto vi mostriamo come realizzare un ottimo emulatore con Raspberry Pi. Il nostro single-board-computer è da sempre stato una scelta ottimale per chi ama l'emulazione. Grazie alla potente CPU del Raspberry Pi 2, adesso è possibile eseguire processi più elaborati e complessi, senza peraltro risentire di alcun calo di prestazioni. Per l'occasione, utilizzeremo l'emulatore **RetroPie** (**http://bit.ly/RetroPieProject**). La scheda SD di cui avrete bisogno per copiare il sistema deve poi essere quanto più spaziosa possibile. Una volta memorizzato RetroPie nella SD, prima di accendere il Raspberry, collegate tutte le periferiche, compresi eventuali gamepad USB. Al primo avvio, RetroPie vi chiederà se volete configurare il joypad per lavorare con l'interfaccia utente (a seconda degli emulatori, cambia il processo di gestione). Se disponete di un controller Xbox 360 cablato, potete consultare l'ottima guida sulla pagina GitHub di RetroPie (**http://bit.ly/RetroPieXbox360SetUp**). RetroPie, come interfaccia predefinita, utilizza **Emulation Station** che permette di emulare una vasta serie di sistemi. Per ognuno di questi, è disponibile un archivio di giochi che viene creato con l'inserimento di una chiave USB nel Pi. A questo punto, uno script genera la struttura dei file necessari su disco.



› ***RetroPie* è in grado di recuperare informazioni dettagliate su ogni gioco. Ha solo bisogno di una connessione al Web**

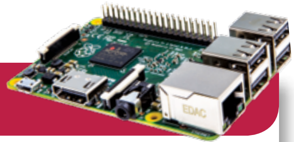


› **L'interfaccia utente di *Retropie* è gestita dalla *Emulation Station* che fornisce un approccio semplice e intuitivo al sistema**

Inserite l'unità nel PC e copiate le ROM dal computer nella rispettiva cartella corrispondente nel drive USB. Adesso tornate su Raspberry e RetroPie copierà automaticamente il contenuto. Aggiungendo le ROM in questo modo, verranno automaticamente abilitate nell'Emulation Station. A questo punto è però importante ricordarvi che alcune ROM sono protette da copyright e quindi rimangono di proprietà dei rispettivi creatori. Il loro utilizzo sotto forma di emulazione rimane comunque in una zona grigia. Molti giochi, infatti, sono talmente vecchi da non essere neppure più disponibili in commercio. Questo però non significa che i diritti d'autore che li proteggono siano scaduti. Cercate quindi di fare molta attenzione a documentarvi in proposito. Una volta installata la ROM, sarà sufficiente tornare nell'interfaccia principale e scegliere il gioco con cui volete divertirvi. RetroPie vi chiederà se volete catalogare i titoli attraverso una connessione a Internet. Se presente, il sistema scandaglierà la Rete alla ricerca di informazioni su ciascun gioco. RetroPie è in grado di emulare parecchi sistemi, tra cui Commodore, Amiga, Sega Genesis e Super Nintendo. Se usate il Pi 2, non c'è alcun bisogno di eseguire un overclock del sistema. Le specifiche hardware sono più che sufficienti per compiere questo genere di emulazione e garantire un'ottima fluidità ai giochi. Se invece usate il Pi originale, potete semplificare la configurazione utilizzando la procedura avanzata che trovate nella wiki di RetroPie (**http://bit.ly/RetroPieAdvConfig**).

Per questo progetto c'è bisogno di...

- » Raspberry Pi 2 modello B (per i risultati migliori)
- » Scheda SD molto capiente
- » Joypad USB
- » Periferiche da usare con il Raspberry Pi 2 come tastiera e mouse wireless



Programmatori storici

A partire dal 1970 fino al 2000 potremmo elencare un'enorme serie di console dedicate ai videogiochi. Aziende come Commodore, Sinclair, Acorn, Dragon, Sega e Atari sono state per anni in competizione. Questa situazione ha quindi permesso a molti programmatori di farsi le ossa, per poi lanciarsi nel ventunesimo secolo con un bagagliaio di esperienze davvero notevole. Uno dei linguaggi più appresi al tempo era BASIC, tornato adesso molto in voga proprio grazie al Raspberry Pi. Infatti, c'è una nutrita comunità di sviluppatori che sperano di sfruttare al massimo le loro competenze proprio con il Pi. David Braben, per esempio, è stato il creatore di **Elite**, uno dei titoli più in voga negli anni ottanta. Si trattava di un gioco di soli 32 K di memoria che ci trasportava nei meandri di una galassia da esplorare. Braben è poi diventato uno dei fondatori della Raspberry Pi Foundation ed è molto attivo nel campo della didattica rivolta ai bambini che si avvicinano al mondo dei computer.

Compatibilità
Tutti i Pi

Primi passi nella robotica

Iniziamo il nostro viaggio verso Skynet. Costruiamo un robot controllato tramite **ScratchGPIO 7** e una scheda **Pibrella**

Per questo progetto c'è bisogno di...

- Un Raspberry Pi 2 modello B o un Raspberry Pi 1 modello A o B
- Pibrella (Cyntech)
- Due motori Micro Gear (Pimoroni)
- Ruote (Pimoroni)
- Telaio (Pimoroni)
- Materiali artigianali di varia natura

Opzionali:

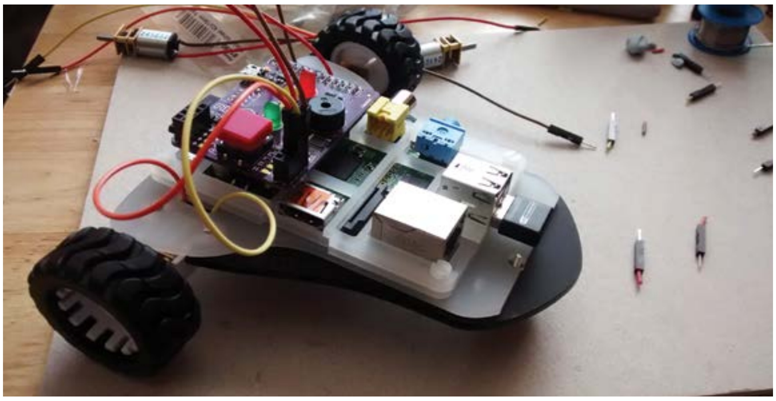
- Batteria USB
- Dongle Wi-Fi

Costruire un robot è diventato un vero rito di passaggio per molti possessori di Raspberry Pi. Grazie a supporti come Scratch GPIO e Pibrella, questo genere di attività è adesso alla portata di molti ed è quindi possibile costruire un automa in modo abbastanza semplice. Nel nostro progetto, vi mostreremo come dare vita a un robot controllato da tastiera o da remoto tramite il progetto VNC. Inizieremo collegando Pibrella al Raspberry Pi. Per connettere la scheda, è sufficiente inserirla nei primi 26 pin del GPIO che si trova vicino alla porta HDMI. Se Pibrella e la HDMI dovessero toccarsi, usate una goccia di argilla in modo da evitare potenziali cortocircuiti. A questo punto, collegate le periferiche al Raspberry, fatta eccezione per l'alimentazione che deve essere connessa direttamente a **Pibrella**. Accendete il Pi e avviate il desktop. Per utilizzare Pibrella avrete bisogno di installare alcuni software, ma prima ancora è obbligatorio connettersi a Internet. In LXTerminal usate i seguenti comandi:

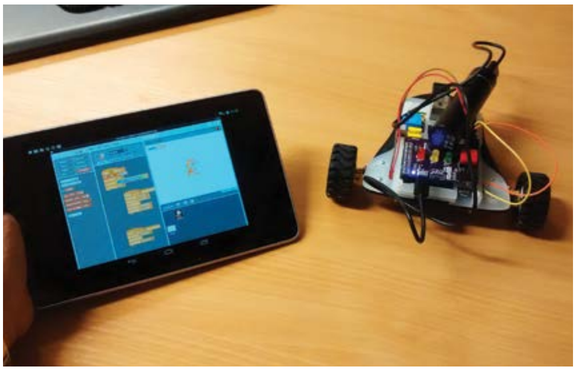
```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install python-pip
sudo pip install pibrella
```

Ora installeremo **Scratch GPIO 7** che utilizza l'interfaccia familiare del linguaggio Scratch, così da consentire a chiunque di muovere i primi passi con il *phisical computing*. Sempre in LXTerminale, scrivete quanto segue:

```
wget http://bit.ly/1wxrqdp -O isgh7.sh
```



› **Basta un po' di materiale e tanta inventiva per creare un semplice automa che si muove a comando tramite la tastiera**



› **Questo progetto può essere gestito anche tramite VNC, così da controllare il robot direttamente da un tablet**

```
sudo bash isgh7.sh
```

Adesso dovreste vedere una Scratch GPIO 7 sul desktop. Avviate l'applicazione e vi troverete di fronte l'interfaccia familiare di Scratch. Per utilizzare Pibrella con ScratchGPIO dovrete prima creare una variabile chiamata **AddOn** e impostarla per essere usata con Pibrella. Infine, create una trasmissione chiamata **AllOff**. Il blocco **Broadcast** è in **Control Palette** e sarà vuoto. Infatti, per funzionare richiede l'input da parte vostra. Per gestire questi blocchi, è opportuno utilizzare il **Green Flag Start**. Ciò comporterà il reset della scheda Pibrella che verrà azzerata non appena il flag è attivo. Questo comporterà quindi un arresto di emergenza per il nostro robot. Ora c'è bisogno di creare il codice che permetterà all'automa di muoversi quando si preme un tasto. In primo luogo, concentratevi sul movimento in avanti che deve essere associato alla freccia verso l'alto. In **Control Palette** trovate un blocco chiamato **When Space Key Pressed**. Inserite al suo interno il vostro codice, quindi selezionate **Up Arrow**. Sotto questo blocco ne verranno creati altri due: **OutputEOn** e **OutputFOn**. Consentiranno di attivare i motori una volta connessi. Sempre da **Control Palette**, connettete il blocco **Wait** e impostatelo per due secondi. Infine, create due nuovi blocchi **OutputEOff** e **OutputFOff** per spegnere i motori. Adesso collegate questi ultimi alle uscite **E** ed **F**, quindi premete la freccia in su. Il vostro robot inizierà così a muoversi in avanti. Per fare in modo che l'automa giri, dovrete solo collegare la ruota sinistra a **E** e la destra a **F**.

Kit per progetti robotici

La robotica è senza dubbio uno dei più interessanti e coinvolgenti progetti a cui ci si possa dedicare. In Rete ci sono diversi kit che possono essere utilizzati per muovere i primi passi. Vediamo quindi qual è il migliore per soddisfare le vostre esigenze. Pibrella che è utilizzato in questo tutorial è estremamente semplice. Non ha sensori e non dispone di alcun meccanismo per la retromarcia. Tuttavia dovrebbe essere un buon punto di partenza su cui focalizzare la propria attenzione. La scheda Pibrella è molto versatile e può essere adattata a molteplici progetti, tra cui la creazione di giochi di società con componenti motorizzate, tipo la **Ruota della Fortuna** (**http://bit.ly/WheelOfFortunePi**). Tramite ScratchGPIO e Pibrella, anche i bambini possono imparare i rudimenti della programmazione, per passare poi più facilmente a Phyton. Un altro kit da prendere in considerazione è **Ryanteck** (**http://bit.ly/RyanTeckBudgetRobotKit**).

Costruiamo un allarme

Compatibilità con Pi 2 o Pi A+/B+

Sfruttando le potenzialità della scheda **Explorer HT Pro**, creiamo un sensore in grado di far scattare un allarme

L'Explorer HAT Pro è uno degli add-on per Raspberry di cui non si dovrebbe fare a meno. Prodotto da Pimoroni, consente di sfruttare un'enorme quantità di funzioni. In questo progetto costruiremo un sensore laser che farà scattare un allarme nel caso venga attivato. Inizieremo installando prima due pacchetti Python, quindi sfrutteremo pip. Per installare pip3, inserite il seguente comando in LXTerminal:

```
sudo apt-get install python3-pip
```

Adesso, aggiungete PyGame e le librerie Explorer HAT:

```
sudo pip3 install pygame
sudo pip3 install ExplorerHAT
```

Una volta fatto, passate alla creazione del codice che sarà alla base del progetto. Potete prelevarne una copia da **http://bit.ly/LXFExplorerHATAlarm**. Per l'occasione, useremo IDLE3 che dovremo lanciare da LXTerminal tramite il comando:

```
sudo idle3 &
```

Ricordate che per utilizzare Explorer HAT pro, sarà necessario avere permessi di root o sudo. Con IDLE3 aprite e create un nuovo file o in alternativa lanciate il codice dal repository GitHub. Le prime tre righe importano le librerie alla base del progetto. Adesso, aggiungete la funzione **sleep** dalla libreria **time**, quindi inserite pygame:

```
import explorerhat as eh
from time import sleep
import pygame
```

La riga successiva inizializza l'audio mixer di pygame che dovrà far funzionare il vostro allarme:

```
pygame.mixer.init()
```

Passiamo adesso al loop principale che useremo per costituire la base per rilevare un'intrusione. Creiamo un loop infinito, quindi una variabile che memorizza lo stato del pin analogico:

```
while True:
  a = (eh.analog.four.read())
```

Successivamente ci accingeremo a utilizzare un **if**, così da controllare l'attivazione dell'allarme. Abbiamo misurato il livello di luce nella nostra stanza che era intorno ai 2,6V. Una torcia che illumina il sensore ha quindi un livello più alto. Abbiamo così deciso di impostare il sensore LDR in modo che qualsiasi cosa sopra i 3V venga considerato OK, così da attivare un LED verde sulla scheda :

```
  if a > 3.0:
    print("SCANNING")
    eh.light.green.on()
    eh.light.red.off()
```

La nostra ultima sezione del codice gestisce l'allarme che si attiva nel momento in cui il fascio di luce prodotto dalla torcia viene interrotto. A ciò dovrebbe poi corrispondere il passaggio dal LED verde al rosso:

```
  else:
    print("ALERT")
    eh.light.green.off()
    eh.light.red.on()
    pygame.mixer.music.load("./alert.mp3")
    pygame.mixer.music.play(1)
```

Una volta completato il software, creiamo l'elemento hardware. Explorer HAT Pro è dotato di un ingresso analogico. Il nostro circuito è comunque piuttosto semplice. Si introducono 5V di potenza verso LDR che viene poi collegato a un resistore 10K. LDR e la resistenza formano un partitore di tensione che viene collegato al pin analogico dell'Explorer HAT Pro. Infine, si collega l'altra estremità della resistenza alla massa (GND).

Adesso avrete solo bisogno di puntare un puntatore laser o una torcia verso la LDR. Assicuratevi di controllare due volte il codice e il cablaggio prima di continuare. Avviate infine il progetto tramite **Run -> Run Module**. La shell IDLE prenderà vita e vi informerà della scansione che sta eseguendo.



› Il progetto comprende una torcia puntata sul nostro sensore. Nel momento in cui il fascio di luce viene interrotto, l'allarme suona

Per questo progetto c'è bisogno di...

- » Un Raspberry Pi modello 1 A+, B+ o Raspberry Pi 2 modello B
- » Un Explorer HAT Pro (Pimoroni)
- » Un LDR (*Light Dependant Resistor*)
- » Resistenza 10K
- » 3x cavi jumper maschio a maschio
- » Torcia o puntatore laser
- » Uno speaker o un monitor con audio incorporato
- » Raspbian OS

Explorer HAT Pro by Pimoroni



L'**Explorer HAT Pro** è l'ultima scheda basata su HAT e prodotta da Pimoroni (**https://shop.pimoroni.com**), una società con sede a Sheffield che produce molti add-on per Raspberry Pi e fornisce un ottimo shop per acquistare accessori di vario genere. Explorer Hat Pro è una scheda all-in-one che introduce nuove tecnologie. Viene fornita con otto touchpad capacitivi che consentono agli utenti di utilizzare i pulsanti sensibili al tocco. Quattro di questi tasti possono poi essere collegati a oggetti conduttivi, come una serie di pinze. Explorer HAT Pro viene inoltre fornito con una serie di ingressi analogici. L'aspetto più interessante di questa scheda è l'inclusione di un controller motorizzato per il pieno supporto dei movimenti in avanti e indietro. Utilizzandolo, potrete sfruttare un'eccellente piattaforma per la maggior parte dei progetti digitali del futuro.

Compatibile con Pi 2 o Pi A+/B+

Una banana come chitarra

Se amiamo musica e cibo, perché non unire le due cose insieme? Usiamo **Capacitive Touch HAT** di Adafruit!

Per questo progetto c'è bisogno di...

- Un Raspberry Pi 2 modello B o un Raspberry Pi 1 modello A+ o B+
- Adafruit Capacitive Touch 12x Sensori HAT
- Morsetti a coccodrillo
- 7 banane

Se vi dicessimo che si può trasformare una banana in una chitarra? Molti ci prenderebbero per folli, eppure è così. Grazie al sensore capacitivo HAT di Adafruit riusciremo in questo originale progetto. Per prima cosa è necessario saldare il connettore a 40 pin sul lato inferiore di HAT in modo che possa inserirsi nella GPIO del Pi.

Una volta fatto, collegate HAT e avviate Raspberry, quindi aprite LXTerminal dove inizierete l'installazione del software necessario:

```
sudo apt-get update
sudo apt-get install build-essential python-dev python-smbus python-pip git
cd ~
git clone https://github.com/adafruit/Adafruit_Python_MPR121.git
cd Adafruit_Python_MPR121
sudo python setup.py install
```

Scaricate il codice che abbiamo adattato per questo tutorial:

```
git clone https://github.com/lesp/LXF_BananaGuitar.git
```

Passate alla directory **LXF_BananaGuitar** e nel terminale lanciate il file **guitar.py** con **sudo idle guitar.py**. Avrete bisogno di importare le librerie necessarie, in questo caso **sys**, **time** e **pygame**. Infine, aggiungete MPR121. Questo chip è ciò che rende il tocco capacitivo possibile.

```
import sys
import time
import pygame
import Adafruit_MPR121.MPR121 as MPR121
```



› **Collegate le banane alla scheda con dei morsetti a coccodrillo, disponendoli nella parte del gambo: musica dalla frutta!**

Successivamente, inizializzate MPR121 e aggiungete un sistema di gestione degli errori che può aiutarvi in caso di problemi:

```
cap = MPR121.MPR121()
# default I2C address (0x5A).  On BeagleBone Black will default to I2C bus 0.
if not cap.begin():
    print 'Error initializing MPR121.  Check your wiring!'
    sys.exit(1)
```

Adesso inizializzate il mixer audio di Pygame:

```
pygame.mixer.pre_init(44100, -16, 12, 512)
pygame.init()
```

quindi create una libreria di suoni da usare:

```
SOUND_MAPPING = {
  0: './1st_String_E.wav',
  1: './2nd_String_B_.wav',
  2: './3rd_String_G.wav',
  3: './4th_String_D.wav',
  4: './5th_String_A.wav',
  5: './6th_String_E.wav',
}
sounds = [0,0,0,0,0,0]
```

Ora configurerete come i vari suoni verranno usati:

```
for key,soundfile in SOUND_MAPPING.iteritems():
        sounds[key] =  pygame.mixer.Sound(soundfile)
        sounds[key].set_volume(1);
```

Infine, create la struttura che permetterà di controllare e verificare se un ingresso è stato attivato:

```
last_touched = cap.touched()
while True:
    current_touched = cap.touched()
    for i in range(7):
        pin_bit = 1 << i
        if current_touched & pin_bit and not last_touched & pin_bit:
            print '{0} touched!'.format(i)
            if (sounds[i]):
                sounds[i].play()
        if not current_touched & pin_bit and last_touched & pin_bit:
            print '{0} released!'.format(i)
    last_touched = current_touched
    time.sleep(0.1)
```

Collegate sei morsetti a coccodrillo agli ingressi da 0 a 5, quindi fissate le clip alle banane. Quando siete pronti, eseguite il codice tramite **Run** -> **Run Module**. Attendete qualche secondo e iniziate a premere il frutto. A ciascuno corrisponderà un suono.

Progetti Touch

Il Touch capacitivo funziona utilizzando materiali conduttivi come frutta, carta stagnola e perfino acqua. Adafruit Capacitive Touch HAT viene fornito con una libreria Python di base che può essere facilmente integrata nei vostri progetti.
Per creare un pianoforte in grado di suonare delle scale, si può utilizzare cavi piuttosto lunghi con i rispettivi morsetti a coccodrillo. Questi vanno poi collegati a dei pezzi di cartone avvolti in carta stagnola che fa da conduttore. Inoltre, è possibile utilizzare HAT come input per dispositivi di scatto remoti. Una banana o qualsiasi altro materiale conduttivo può essere sfruttato per far scattare una fotocamera. Questa scheda è abbastanza nuova sul mercato e, nel momento in cui scriviamo, ci sono solo poche risorse disponibili per sfruttarla al meglio. Si tratta comunque di un dispositivo geniale, capace di dare vita a progetti originali.

Raspberry Pi psichedelico

Compatibile con Pi 2 o Pi A+/B+

Siamo amanti della discoteca? Allora creiamo degli effetti luminosi psichedelici con **Unicorn HAT 8x8 RGB LED**

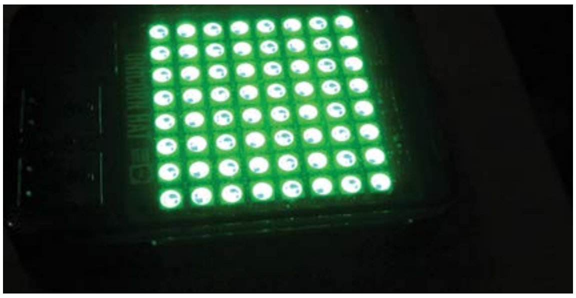
Unicorn HAT è un add-on per Raspberry PI modelli B+, A+ e Raspberry Pi 2. Volendola descrivere, si tratta di una griglia da 8x8 di NeoPixel LED super luminosi. Lo stesso tipo di LED viene utilizzato per svariati scopi, ma soprattutto dove c'è bisogno di ottenere la massima luminosità con il minimo consumo di energia. Per l'occasione creeremo due script. Per installare Unicorn HAT in Raspberry Pi, collegatelo con fermezza su tutti i pin del connettore GPIO. Una volta fatto, connettete le periferiche, assicurandovi di avere una connessione a Internet. Infine, avviate il Pi e di conseguenza accedete al desktop. Attenzione, i LED NeoPixel sono particolarmente brillanti, quindi proteggete la vista con un pezzo di carta per evitare di rimanere abbagliati. Adesso dovrete installare il software necessario. Aprite LXTerminal:

```
sudo apt-get install python3-pip python3-dev
sudo pip-3.2 install unicornhat
```

Mantenete aperto il terminale e scrivete **sudo idle3 &**, così da lanciare IDLE3 nell'editor Python 3 con permessi sudo. In questo modo, potrete utilizzare GPIO. Con IDLE3 aperto, avrete a disposizione la shell Phyton, dove invierete i comandi direttamente. Andate nel menu **File** e fate click su **New** per aprire un documento vuoto. Per il nostro progetto creeremo una sequenza che consente di cambiare colore passando dal rosso al verde, finendo poi con il blu in un ciclo infinito. Prima di tutto, importate le librerie di cui avete bisogno:

```
from time import sleep
import unicornhat as u
```

Come vedete, abbiamo aggiunto la funzione **sleep** solo alla libreria **time**, così da risparmiare risorse di sistema.



› **Il nostro primo progetto ci ha permesso di creare una griglia di un solo colore che poi varia con un altro**



› **Estendendo le potenzialità del progetto base, siamo riusciti a realizzare una caleidoscopica luce da discoteca**

Nella riga successiva, importiamo la libreria Unicorn HAT e rinominiamola con **u** per lavorarvi più facilmente. Adesso passiamo al corpo principale del codice contenendolo in una struttura **try-except**

```
try:
    while True:
        for i in range(8):
            for j in range(8):
```

Usiamo **while True** per creare un ciclo infinito, al cui interno abbiamo rispettivamente due cicli: **i** e **j**. Questi contengono un valore da 0 a 7 che controllerà la posizione sulle assi X e Y della griglia a LED. Ora create le azioni che verranno compiute durante l'uso:

```
                u.brightness(1.0)
                u.set_pixel(i,j,255,0,0)
                u.show()
                sleep(0.01)
```

In primo luogo, abbiamo impostato la luminosità al massimo. I colori sono gestiti tramite un mix di rosso, verde e blu come segue: 255,0,0 rosso, 0255, 0 verde e 0,0,255 blu. Infine, abbiamo dovuto specificare ad HAT che la variazione cromatica deve avere un intervallo di 0.01 secondi e abbiamo fermato il codice per 2 secondi prima di passare al prossimo ciclo per la gestione del verde e poi del blu. L'ultima sessione permette di interrompere il loop premendo **Ctrl+C**. Avviate il progetto con **Run** -> **Run Module**. Per il codice già pronto, collegatevi a **GitHub** (**http://bit.ly/LXFUnicornHAT**). LXP

Per questo progetto c'è bisogno di...

» Un Raspberry Pi modello 1 A+ B+o un Raspberry Pi 2 B
» Unicorn HAT (Pimoroni)

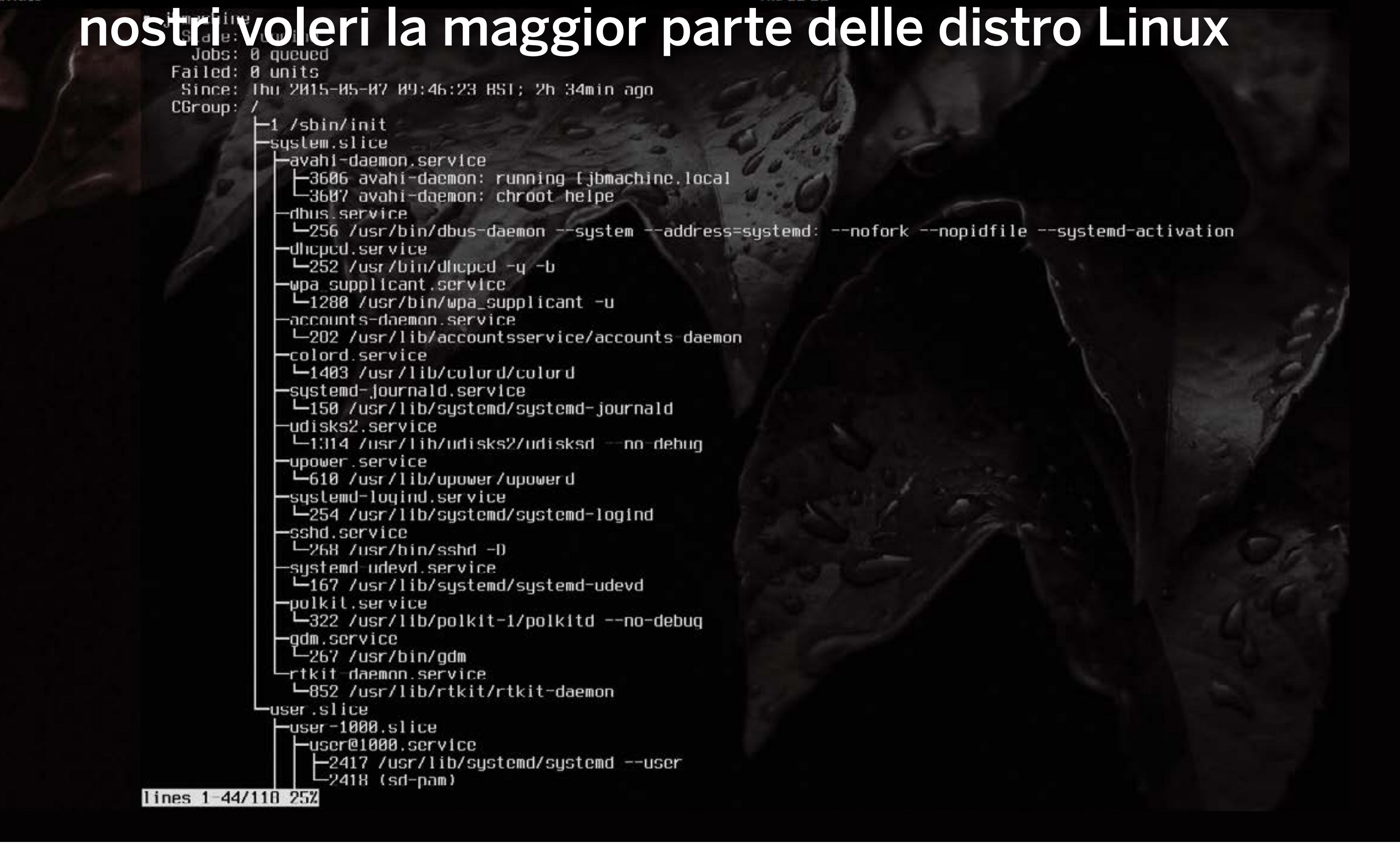
La potenza dei NeoPixel



Unicorn Hat è stata la prima scheda HAT progettata da Pimoroni per il Raspberry Pi modello B+. HAT (*Hardware Attached on Top*) è uno standard che utilizza una EEPROM (*Electrically Erasable Programmable Read Only Memory*) per comunicare con il Pi e configurare in automatico la scheda pronta all'uso. Recentemente abbiamo avuto la possibilità di parlare con Pimoroni di questo add-on. Ne è emerso che il progetto alla base di Unicorn è stato portato avanti per meglio approfondire lo studio del nuovo standard HAT. I LED di Unicorn sono conosciuti anche come NeoPixels o almeno questo è il nome che Adafruit utilizza per descriverli. La loro denominazione corretta è un'altra: **WS2812**. I NeoPixel stanno trovando sempre più campi d'uso, soprattutto là dove c'è bisogno di molta più luminosità rispetto a quella di un normale LED.

Systemd

Scopriamo i segreti di Systemd per piegare ai nostri voleri la maggior parte delle distro Linux



Da quando Systemd è diventato il sistema init predefinito di Fedora 15 nel 2011, molte altre distribuzioni hanno iniziato ad adottarlo. In definitiva, dopo le ultime versioni di Debian e Ubuntu, solo Gentoo e Slackware continuano a sfruttare SysVinit. Naturalmente ci sono diverse distribuzioni più piccole che non aderiscono ancora a Systemd. Tuttavia, il fatto che molti utenti continuino a utilizzare le piattaforme che lo hanno adottato senza peraltro lamentarsi, significa che il suo funzionamento si è dimostrato stabile e flessibile. Uno dei vantaggi di Systemd riscontrato da molte persone, è il tempo di avvio del sistema. Non solo, ma grazie alla parallelizzazione dei servizi e al modo in cui questi vengono unificati, nonché alla presenza di svariati script, Systemd si è rivelato un'ottima scelta anche per gli amministratori di sistema. Tuttavia, in quest'ultimo periodo sono state aggiunte diverse novità che tuttavia sono per lo più passate in sordina. Infatti molti utenti non sono ancora in grado di sfruttare Systemd nel pieno delle sue funzioni. In queste pagine, pertanto, ci proponiamo lo scopo di approfondire l'uso e la configurazione di questo gestore di sistema, così da permettervi di utilizzarlo senza problemi.

> "Systemd, grazie alle sue caratteristiche, è un'ottima scelta per i SysAdmin"

Systemd è un gestore di sistema che ha come modo di operare primario quello di init. Il suo binario principale è un link simbolico al file /sbin/init che viene gestito come Process ID (PID) 1 dopo il caricamento del kernel. Systemd provvede quindi al caricamento di tutti i servizi e alla loro supervisione fino alla disattivazione. Aspetto, questo, che in linea di massima lo può accomunare alla definizione di padre di tutti i servizi. Il precedente sistema init, noto come SysVinit e nato in System V (una prima versione di Unix), può essere definito come poco più di un'obsoleta raccolta di script tenuta insieme da **greybeard**. SysVinit ha funzionato abbastanza bene fino a quando le distro Linux non si sono evolute oltre un certo limite. Una volta superato, le peculiarità di questo sistema hanno iniziato a fare i conti con i propri limiti che non riuscivano più a soddisfare le richieste delle moderne distribuzioni. Così, nel 2006, Canonical ha iniziato a svilupparne l'evoluzione, conosciuta con il nome di **Upstart**. Quest'ultimo era retro-compatibile con le vecchie versione di SysVinit, ma al contempo riusciva a fornire una maggiore capacità di gestione delle dipendenze e una migliore interazione con la tecnologia asincrona. Oltre a Ubuntu, Upstart è stato adottato anche da tutte le distribuzioni Red Hat, così come da Chrome OS. Tuttavia, nonostante ciò, dal 2013 quasi tutte le distro hanno iniziato a sfruttare Systemd. Nel 2014, il Debian Technical Commitee ha votato per passare in toto a quest'ultimo gestore di sistema.



› *Devuan* è una fork di Debian che rifugge Systemd. È però ancora in uno stato pre-alpha. Se volete una distro senza Systemd, puntate su *Slackware*, *Gentoo* o *PCLinux OS*

Seat e sessioni

Uno dei motivi che hanno portato all'adozione di Systemd, è la sua capacità di unificare le funzioni incentrate sul desktop. Il suo componente **logind** formalizza il concetto di seat, sessioni e utenti. In questo modo, utilizzando l'hardware adatto, la gestione delle sessioni desktop locali in simultanea diventa un'operazione davvero semplice. Anche se non tutti possono ancora beneficiare di tale aspetto, c'è però un'ulteriore considerazione da fare. Con tale funzione, infatti, la vecchia logica alla base di **ConsoleKit** diventa del tutto obsoleta. Tornando indietro nel tempo, chi non utilizzava un ambiente desktop completo, avrebbe dovuto combattere non poco per montare una semplice chiavetta USB o spegnere il sistema senza richiedere i privilegi di root. Con **systemd-logind** anche il server X può essere eseguito come un utente, così da incrementarne la sicurezza.
Al contrario, però, gli ambienti desktop come Gnome hanno iniziato a fare affidamento sui componenti di Systemd, i quali non sono sempre semplici da installare. I comandi **reboot, halt** e **shutdown** richiedono tutti l'uso di root. Tuttavia, systemd-logind (insieme al pacchetto **polkit**) permette a queste funzioni di essere svolte localmente da qualsiasi utente connesso con una sessione X attiva.
Tale utente, quindi, sarà in grado di spegnere il computer con:

```
$ systemctl poweroff
```

a condizione, naturalmente, che nessun altro account sia connesso. Nel caso ve ne sia uno, sarà comunque richiesta la password di root. È poi possibile sostituire il comando **poweroff** con quelli destinati alla sospensione o all'ibernazione del sistema (se questo lo prevede). Systemd-logind gestisce anche diversi pulsanti che tradizionalmente venivano supervisionati da **acpid**. La loro configurazione è presente in **/etc/systemd/logind.conf** che fornisce i seguenti valori di default autoesplicativi:

```
IdleAction=ignore
HandlePowerKey=poweroff
HandleSuspendKey=suspend
HandleHibernateKey=hibernate
HandleLidSwitch=suspend
HandleLidSwitchDocked=ignore
```

Journal interno

Un altro servizio non più necessario con l'adozione di Systemd è **syslog** (è tuttavia vero che Systemd può inoltrare i messaggi a un syslog daemon nel caso sia richiesto).
Il demone **journald** di Systemd, infatti, si dimostra più che sufficiente per soddisfare le necessità di log relative a ogni utente. Prima di journald, i messaggi venivano raccolti dal kernel e qualsiasi esecuzione (o mancanza) era invece annotata in syslog. Questo, a tal proposito filtrava le varie comunicazioni in file »

La vita senza Systemd

Alcune distribuzioni, nonostante utilizzino Systemd per impostazione predefinita, vi permetteranno di usare un altro sistema di init alternativo. Il supporto varia in base alla distro. Ubuntu 15.04, per esempio, rende il processo molto semplice e permette di usare sia Systemd sia Upstart. Per selezionare il gestore desiderato, basterà entrare nelle opzioni avanzate del sotto-menu **Ubuntu Grub**. Coloro che sono alla ricerca di uno switch capace di installare il pacchetto parvenu-sysv, possono usare:

```
$ sudo update-initramfs -u
```

Per ora, la maggior parte degli utenti Ubuntu non si è lamentata di alcuna difficoltà. Tuttavia Systemd è comunque destinato a entrare a far parte in pianta stabile dell'ecosistema di Ubuntu. Basterà aspettare il rilascio LTS del prossimo anno. Un altro sistema init di cui vale la pena parlare è **OpenRC**. Anche se tecnicamente non può essere considerato un sostituto di SysVinit, permette di avere una visione più ampia su tutto ciò che è accaduto dopo PID1. OpenRC viene mantenuto e utilizzato per impostazione predefinita in Gentoo. Dal momento che udev è destinato a fondersi in Systemd, gli utenti che usano questo sistema dovranno sfruttare eudev. Comunque non c'è niente di cui preoccuparsi. Infatti, è possibile utilizzare sia OpenRC sia eudev in altre distro come Arch Linux.

» di testo presenti in **/var/log**. Per evitare che questa directory diventasse enorme, di solito si provvedeva a installare e configurare **logrotate**. Con Systemd, invece, tutti i registri sono centralizzati e accessibili tramite il comando **journalctl**. Naturalmente, se avete ancora bisogno di un'implementazione di syslog, niente vieta di eseguirlo in concomitanza con journalctl. Quest'ultimo, è importante ricordarlo, gestisce anche i registri più lontani nel tempo. Essi non vengono eliminati, a meno che lo spazio su disco non vada oltre una certa soglia indicata nel file **/etc/systemd/journald.conf**. A tal proposito, di seguito trovate tre opzioni che potreste voler modificare:

» **SystemMaxUse** specifica lo spazio massimo occupabile su disco dal journal. Il valore predefinito è il 10% del filesystem.

» **SystemKeepFree** specifica lo spazio minimo che Systemd cercherà di mantenere libero sul filesystem occupato dai log. Se questo è superiore allo spazio disponibile, il valore viene regolato in base alla quantità disponibile rilevata all'avvio di Systemd.

» **SystemMaxFileSize** indica la dimensione massima di ogni singolo file journal.

In definitiva, questo specifica a Systemd il numero massimo di file che possono danneggiare il log. Nel caso avessimo bisogno di controllare i registri a partire da oggi, basterà utilizzare lo switch **-b** che mostra i soli messaggi a partire dall'avvio corrente. Ogni volta che qualcosa non funziona, di solito un utente Linux prova a controllare l'output di:

```
$ dmesg | tail
```

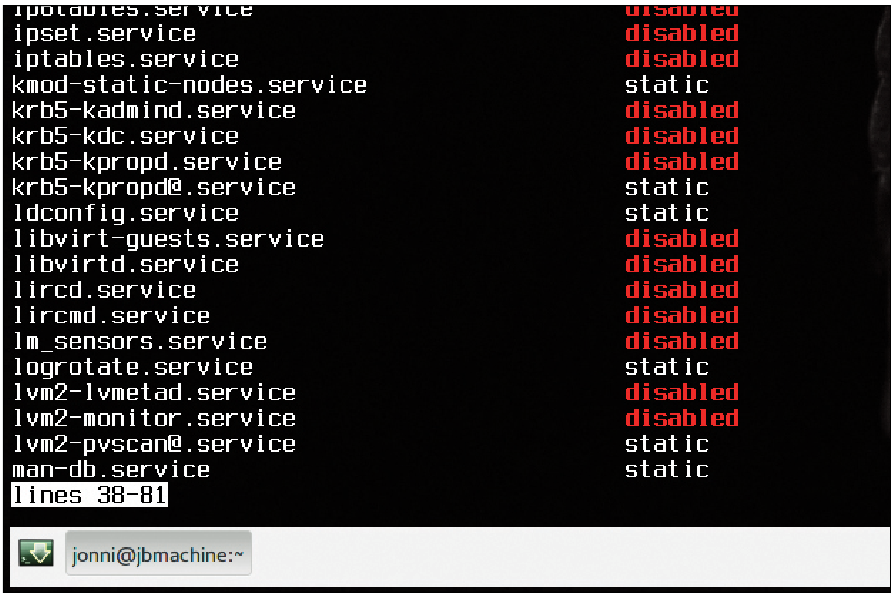
che mostra eventuali messaggi di errore provenienti dal kernel.

```
$ tail /var/log/messages
```

invece mostra quelli derivanti da altre funzioni. L'equivalente di Systemd è molto più semplice e veloce:

```
$ journalctl -e
```

Questo comando permette di scorrere il file di log dalla fine fino al principio. Naturalmente, **dmesg** funziona ancora, ma in questo modo vediamo solo i messaggi provenienti da fonti al di fuori del kernel. Inoltre, il timestamp è regolato automaticamente in ora locale, anziché a partire dal boot di sistema. Se qualcosa è andato storto durante un avvio precedente, potete quindi controllare i registri con l'aggiunta di un numero allo switch **-b**. Usando **-1**, per esempio, vi riferirete all'avvio corrente, **-2** a quello precedente e così via. È inoltre possibile utilizzare l'indicizzazione assoluta. 1, quindi, si riferirà al primo avvio nel log di Systemd.



› I file *unit* sono ovunque. Possono essere definiti la linfa vitale di Systemd

Il dibattito sui binari

I registri di systemd sono memorizzati in formato binario, così da facilitarne l'indicizzazione. Questo vi consente di eseguire ricerche estremamente rapide, nonostante la quantità ingente di materiale presente. I registri binari, tuttavia, sono più inclini alla corruzione. In teoria, quindi, un guasto del disco che colpisce un settore da 4 k di un file di testo potrebbe danneggiare la totalità di un binario journald. I file di testo si prestano poi ad analisi con Perl, grep, sed, awk e simili e molti amministratori di sistema fanno uso di script che li incorporano per lavorare meglio con i log. Il fatto quindi che gli script non vengano più utilizzati, ha generato un certo malumore nella comunità dei SysAdmin. Ciò nonostante, dal nostro punto di vista, pensiamo che questa critica sia ingiustificata. Se avete bisogno di file di testo, basterà utilizzare nuove versioni di **syslog-ng** che li estrarrà senza problemi da journald. Le unità fondamentali di Systemd vengono appunto chiamate **unit**. Il comando:

```
$ systemctl list-unit-files
```

permette di visualizzarne un elenco completo, mostrando anche il loro status. I file unit si trovano nelle sottodirectory **/system** o **/user/** della directory principale di Systemd (di solito **/usr/lib/systemd**). I file unit possono essere servizi (per esempio, **sshd.service**) che inizializzano programmi, demoni o simili. In alternativa, possono anche essere elementi

Systemd: cosa c'è che non va?

Una delle critiche più feroci fatte a Systemd dai suoi acerrimi detrattori è la sua presunta violazione delle filosofie Unix tradizionali. Questo, infatti, è accusato di essere un blob monolitico che usurpa (tra gli altri) udev, cron, PAM, acpid e logind. Tutti questi componenti, trovandosi in un solo binario come PID1, sono stati per anni alla base degli strumenti per amministratori di sistema. Secondo la nostra opinione, questo modo di pensare è pressoché infondato e per certi versi decisamente sorpassato. Il fatto che Systemd abbia trovato il favore delle distro che lo stanno usando sempre di più dovrebbe far comprendere come la sua attività apporti effettivamente un notevole miglioramento all'esperienza d'uso. I più tradizionalisti, sotto questo profilo, continuano a insistere su un possibile complotto che vede gli sviluppatori come responsabili di imporre agli utenti le loro preferenze. Systemd, tuttavia, fornisce tutto quello di cui si può aver bisogno per gestire un sistema moderno: ambienti chroot migliorati (tramite systemd-nspawn e machinectl), tempi di avvio più rapidi e molto altro ancora. Certo, il suo utilizzo non è dei più immediati e può spaventare gli utenti alle prime armi, ma almeno permette di cimentarsi con una sintassi moderna. Naturalmente, Systemd è ancora molto giovane e come tale soffre di alcune mancanze che comunque dovrebbero essere colmate nel minor tempo possibile.

molto più astratti, come punti di mount, dispositivi, target o altro. I target sono un'interpretazione più flessibile del runlevel di SysV e definiscono un insieme di servizi di partenza che si occupano di una funzione particolare. I sistemi desktop, per esempio, si avviano all'interno del target **graphical** predefinito che corrisponde a runlevel 5. I server, invece, possono eseguire il boot con **multi-user.target**, analogo al runlevel 3. Se esaminate **graphical.target**, potrete notare le seguenti righe:

```
Requires=multi-user.target
Wants=display-manager.service
```

Questo vi informa che il target **graphical** comprende qualsiasi cosa riguardante il target multi-utente, ma ha anche bisogno di un display manager da caricare. Il sistema, a questo punto, può essere forzato per usare un particolare target (solo con privilegi di root) tramite:

```
$ systemctl isolate multi-user.target
```

Il file **display-manager.service** è in realtà un link simbolico che viene impostato al momento dell'installazione del display manager. I servizi vengono quindi aggiunti ai target Systemd mediante il comando `$ systemctl enable`, il quale crea solo i simlink necessari. Per lanciare il demone SSH all'avvio successivo, eseguite:

```
$ systemctl enable sshd
```

Sarete così informati sulle azioni di Systemd:

```
Created symlink from /etc/systemd/system/multi-user.target.wants/sshd.service to /usr/lib/systemd/system/sshd.service.
```

Quando le cose vanno male

Anche se nella maggior parte dei casi tutto andrà per il meglio, è inevitabile che talvolta ci si trovi ad affrontare qualche guaio. Quando un sistema si corrompe, molto spesso non si avvia. In tal caso, la colpa è quasi sempre imputabile ai driver video. Il sistema, avendo omesso di avviare il gestore grafico per il login, vi restituirà un'ineluttabile schermata nera. Per ovviare a questo inconveniente, sperando che la macchina vi permetta ancora di operare, basta aggiungere la seguente opzione alla riga di comando del kernel (premete **e** per modificare il menu di Grub)

```
systemd.unit=multi-user.target
```

Avviando con questa opzione (premendo **Ctrl**+**X**), impedirete al sistema di impantanarsi nel caricamento del gestore grafico di login, permettendovi così di risolvere i problemi da riga di comando. Per altri malfunzionamenti che potrebbero ostacolare l'avvio non è da escludere il ricorso ai target di soccorso o, in casi estremi, all'uso di **chroot-ing** da un altro sistema operativo. Naturalmente, non tutto ciò che si danneggia può provocare un mancato avvio. Spesso, potrebbero comparire strani messaggi di errore o in alternativa fastidiosi blocchi nella procedura di boot. Oltre a dare un'occhiata al journal, è quindi possibile ottenere una panoramica sulla salute del sistema con:

```
$ systemctl status
```

Questo comando mostra i processi in coda ed elenca i file attualmente in esecuzione. Se nella seconda riga leggete:

```
# State: degraded
```

(evidenziato in rosso) significa che qualcosa non funziona. Di solito, è colpa di un file unit che non viene caricato. Per approfondire, quindi, usate semplicemente:

```
$ systemctl --state=failed
```

Una volta che l'unit responsabile è stato identificato, utlizzate `journalctl` per controllare che tutte le informazioni utili al journal siano ancora presenti. Per esempio, se il comando sopra riportato mostra qualche errore con **sshd.service**, potete interrogare il journal per capire quale sia stato l'ultimo processo che ha interagito con quel dato servizio. Basta usare il comando seguente:

```
$ journalctl -eu sshd
```

Questo, si spera, vi fornirà informazioni sufficienti per risolvere il problema. Riavviate quindi il servizio con:

```
$ systemctl restart sshd
```

A questo punto, se tutto va per il meglio, lo stato cambierà da un preoccupante **degraded** a un più tranquillizzante **running**.

> **"Le unità fondamentali di Systemd vengono chiamate con il termine unit"**

Alcuni processi userspace possono scrivere nel journal. Nel caso, potete filtrare la ricerca con il nome del servizio (usate l'opzione `_COMM=`), percorso assoluto o PID (`_PID=`). Da Gnome 3.12, i log di X.org non vengono più scritti nel file **/var/log/Xorg.0.log**. Al contrario, risiedono nel journal ed è possibile filtrarli con:

```
$ journalctl -e _COMM=Xorg
```

o tramite:

```
$ journalctl -e /usr/bin/Xorg
```

Se state usando Gnome su Fedora o Arch Linux, allora dovrete utilizzare `Xorg.bin` o `gdm-x-session` in _COMM appena analizzato.

Boot veloce

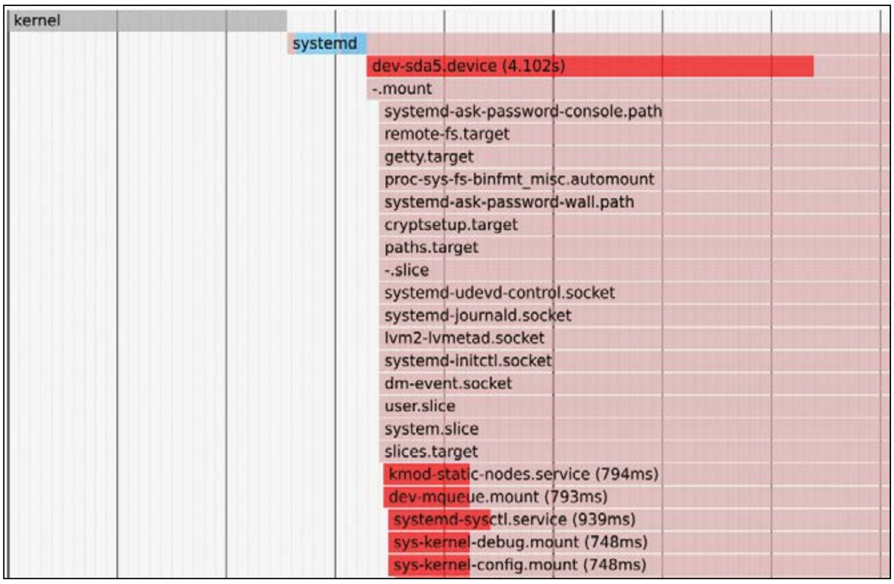
Una caratteristica decisamente piacevole di Systemd è la sua capacità di velocizzare i tempi di avvio. L'istruzione

```
$ systemd-analyze
```

vi mostrerà un pratico riepilogo di quanto riuscite effettivamente a incrementare le prestazioni del boot del vostro sistema. Per maggiori dettagli, aggiungete semplicemente `blame` al comando, così da avere la panoramica completa dei tempi di lancio di ogni singolo servizio. Tenete presente però che in questo caso la visualizzazione richiederà effettivamente un po' di tempo. In più, verrà mostrata con una scaletta che partirà dai servizi che hanno impiegato di più per avviarsi. In alternativa, si può anche utilizzare:

```
$ systemd-analyze plot > plot.svg
```

il quale mostra tutte le informazioni sulle tempistiche di avvio. LXP

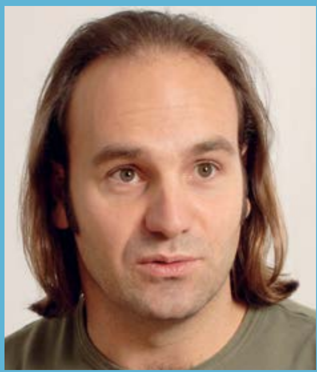


› **Con il comando *systemd-analyze* otterrete una panoramica puntuale e precisa dei tempi di avvio del computer. Vedrete così quanto Systemd incide positivamente sul boot**

Open Source Heroes









Negli anni abbiamo parlato con i più grandi nomi nel mondo dell'Open Source... Ecco il meglio delle nostre chiacchierate



Richard Stallman
Creatore di GNU
Estate 2011

Linux Pro: La maggior parte dei nostri lettori è decisamente appassionata dall'argomento del Software Libero...
RMS: Ma usano specificamente il termine Software Libero o Free Software?

LXP: Beh, quando ci scrivono i nostri lettori usano a volte il termine 'Software Libero' e a volte l'espressione 'Open Source'...
RMS: Ah, però sono due cose diverse. Open Source si riferisce a diverse idee e filosofie. La differenza è fondamentale, perché è a livello di valori. Non è una disputa su qualche dettaglio: è un disaccordo su un principio di fondo. Il nostro obiettivo è una società libera, in cui gli utenti hanno completa libertà. Le organizzazioni e i leader del mondo Open Source dichiarano che il loro obiettivo è ottenere una miglior qualità del codice. Questi due scopi non potrebbero essere più distanti: per noi il punto sono la libertà e la solidarietà sociale, per loro la qualità della programmazione.

LXP: Non è un modo per far passare la gente da un sistema a un altro, però?
RMS: Non capisco: sono diversi.

LXP: Per esempio, se hai un'azienda che fa software proprietario, potrebbe risultare difficile adattare il modo di pensare al GPL e al Software Libero. Se però riesci a farla passare all'idea di Open Source, grazie ai benefici che può portare alla qualità, una volta che si è abituata a questo modo di pensare può spingersi oltre...
RMS: Non sono la stessa cosa e il primo di solito non porta al secondo. In realtà, quando la filosofia dell'Open Source si espande molto, come ha già fatto, tende a rendere le persone più chiuse nei confronti dell'idea del Software Libero. Porta persino a far passare inosservata la nostra stessa esistenza. La maggior parte degli articoli che tratta di GNU non lo chiama sistema GNU o Software Libero. Lo descrivono come Open Source e danno l'impressione che noi, gli sviluppatori, seguiamo le idee dell'Open Source di cui i lettori hanno già sentito parlare: non immaginano nemmeno quali sono i nostri reali obiettivi.

LXP: Alcune persone, quando sentono parlare di 'Free Software' pensano a dello spyware per computer con Windows.
RMS: Ho impiegato degli anni a rendermi conto che questa distinzione è vitale. Nel 1983, quando ho annunciato GNU, non avevo separato i due concetti. Ci sono voluti un po' di anni perché arrivassi a farlo. Ancora nel Manifesto di GNU, pubblicato nel 1985, non è chiara la distinzione tra i due significati della parola 'free'. Dopo questa data mi sono reso conto che è necessario enfatizzare che con Free si intende Libero, non gratuito. Sarebbe stato meglio che me ne rendessi conto prima, ma anche in questo caso non avrei saputo bene cosa fare perché in inglese, la mia lingua, free significa sia "libero" sia "gratuito" e non c'è

modo di usare una parola specifica per "libero". Per superare questo equivoco abbiamo iniziato a usare l'espressione 'free/libre' che rende subito chiaro il punto in questione. Per rendere più chiara la distinzione tra Free Software e Open Source possiamo usare l'esempio di come è nato **LibreOffice**. Sun ha acquisito **StarOffice** e lo ha pubblicato come Free Software con il nome di **OpenOffice.org**. I responsabili di Sun che hanno gestito questa operazione, però, non supportavano dal punto politico le idee del Free Software, perché erano in realtà interessati all'Open Source. Il loro scopo era di ottenere un buon livello di qualità e successo per il loro prodotto, non di dare la libertà agli utenti. La libertà non era il loro obiettivo: anche se il loro codice sorgente era libero il loro modo di pensare non era quello del Free Software ma dell'Open Source. Così hanno creato una lista di estensioni includendone alcune proprietarie. Come potevo reagire? Abbiamo deciso di chiedere la collaborazione della comunità per creare la nostra lista di estensioni. LibreOffice usa le nostre estensioni. Il problema è stato risolto grazie al fatto che le persone che curano questa versione del programma sono attivisti della filosofia del Free Software: il loro obiettivo è la libertà e questa attitudine guida le loro scelte. Questo esempio dimostra che le persone che non pensano alla libertà o non le danno valore e a volte fanno scelte che vanno in direzioni diverse.

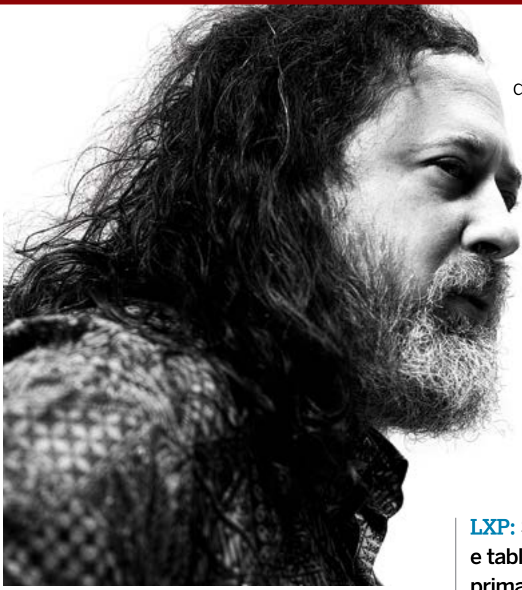
LXP: Vediamo varie distribuzioni GNU/Linux completamente free, come Trisquel e gNewSense, ma molte di esse hanno dei problemi, sono progetti molto dispersivi. C'è spazio per uno GNU ufficiale? GNU/Linux di GNU?

RMS: Secondo me sarebbe meglio se riuscissero a collaborare di più tra loro. Non voglio creare un'altra distro GNU perché sarebbe una mancanza di rispetto per tutte le persone che stanno lavorando sulle distro attuali. Non mi piace prendere una posizione a favore di uno o dell'altro, avere una preferenza. Mi sembrerebbe una scelta infelice.

LXP: Saremmo curiosi di sapere con cosa lavori. Sul tuo sito ci sono foto di te all'opera con un ThinkPad ma attualmente non lo consigli.

RMS: Non uso il ThinkPad, le foto sono di vecchia data. Ora uso un dispositivo di Lemote machine: Yeeloong. L'ho scelto perché è tutto free, fino al BIOS. Ha un processore con architettura di tipo MIPS, un suo equivalente cinese. Il punto sostanzialmente è che fa quel che serve.

LXP: Tornando a una visione più d'insieme, secondo te qual è la principale attuale minaccia per il Free Software?



RMS: Ce ne sono diverse. Ci sono limitazioni a livello legale, come brevetti software in alcuni Paesi con politiche folli. Ci sono inoltre leggi che penalizzano esplicitamente il Software Libero, come il Digital Millennium Copyright Act negli Stati Uniti, che proibisce il Free Software con cui si possono superare sistemi di limitazione digitale che impediscono le modifiche del codice. L'Unione Europea ha leggi simili. Sia gli Stati Uniti sia l'Unione Europea cercano di spingere leggi come queste anche in altri Paesi, attraverso trattati. Quindi ho una cattiva considerazione di questi governi. Poi ci sono gli ostacoli creati dai produttori, che spesso lavorano con Microsoft. Per esempio, ci sono molti componenti hardware per PC che si possono usare solo da Windows. Tipicamente le specifiche di quei componenti non sono disponibili, quindi ci troviamo chiaramente di fronte a una pratica commerciale non etica: vendere a qualcuno un prodotto e rifiutarsi di dirgli come si usa. Non dovrebbe essere premesso. Un altro ostacolo è la tendenza a vendere computer in bundle con Windows. Se fosse per me proibirebbero anche quella pratica. Poi c'è la tendenza delle aziende a donare alle scuole copie gratuite o quasi dei loro programmi a pagamento. Microsoft e Apple lo fanno e ho letto che è una pratica seguita anche dalla Gates Foundation. L'idea di beneficienza di Bill Gates è di assuefare gli studenti scolastici a Windows, in modo che lui stesso possa fare più soldi. Non mi sembra beneficienza.

LXP: Una domanda filosofica: è ancora possibile creare un mondo in cui tutto il software è libero? Dovrebbe ancora essere questo l'obiettivo finale?

RMS: Sì, secondo me dovrebbe essere l'obiettivo finale. Potrebbe risultare impossibile sradicare completamente gli ultimi stralci di software non libero. Dopo tutto, in quasi duecento anni di abolizionismo non abbiamo ancora eliminato la schiavitù. Ci sono posti in cui le persone sono in pratica degli schiavi. Ho letto dichiarazioni di lavoratori stranieri nel Regno Unito che sono a tutti gli effetti in schiavitù, perché se si lamentassero finirebbero con l'essere deportati. È difficile eliminare totalmente alcune forme di abuso ma sono sicuro che si possa raggiungere una società in cui il software proprietario sia considerato un'eccezione particolare: dobbiamo però pretenderla insieme.

LXP: Sempre più persone usano smartphone e tablet come piattaforme informatiche primarie, con i loro centri vendita di app...

RMS: Uno smartphone è un computer: la funzione che svolge è esattamente quella. Di conseguenza, tutto ciò che diciamo sui computer e il fatto che il software che impiegano dovrebbe essere libero, vale anche per i dispositivi mobili, tablet inclusi. Ora, cosa dire dei centri vendita di app? Prima di tutto, quelli di Apple e Microsoft non accettano Software Libero. Questo dimostra quanto sono malvagi. Per Android il discorso è diverso. Il codice sorgente di Android è gratuito e pubblicato da Google, ma non adotta una licenza copyleft tranne che nel caso di Linux che è sotto licenza GPL v2. Il risultato è che la licenza non protegge gli utenti dalla "Tivoization", cioè la pratica di rendere l'esecuzione di un programma libero in realtà non libera, proibendo all'utente di usare una sua versione personale.

LXP: Adottano una strategia simile anche i miei genitori... si tratta di trovare l'approccio giusto.

RMS: Per spiegare la situazione uso l'esempio delle ricette. È una buona analogia perché un programma sotto diversi aspetti assomiglia a una ricetta. Entrambi consistono in una serie di passaggi da seguire per ottenere il risultato desiderato. Se consideri come i cuochi utilizzano le ricette, noterai che in pratica godono delle quattro libertà fondamentali: le cucinano liberamente, le possono studiare e cambiare quando vogliono, le alterano come desiderano e, se creano una versione modificata di una ricetta, possono distribuire liberamente delle copie della loro versione. Immaginati cosa succederebbe se le aziende e lo stato cercassero di imporre delle ricette proprietarie. Supponiamo che lo stato dicesse: da domani, se copi o cambi una ricetta ti mettiamo in prigione per pirateria. Lo stato non applica questa politica alle ricette, ma è quello che hanno cercato di fare con il software. »



Linus Torvalds
Creatore di Linux
Novembre 2012

LXP: Il desktop Linux ha fallito perché c'è troppa scelta? [Ridendo]
Linus Torvalds: Non mi sembra che il desktop stia andando troppo bene e ci sono dei motivi tecnici. Avrai probabilmente visto le mie tirate in cui affermo che, sotto certi aspetti, il desktop stia andando nella direzione sbagliata ma la ragione principale consiste nel fatto che la maggior parte delle persone "normali", non appassionate di informatica, non vuole installare un sistema operativo. Un desktop non può funzionare se non è preinstallato. Ci sono casi in cui, sapendo dove guardare, potresti trovare Linux pre-installato in alcuni computer Dell. Realisticamente, però, non ci sono computer con Linux preinstallato.

LXP: Cosa sarebbe successo, però, se i team di KDE e Gnome non avessero fatto tanti sforzi per creare le prime versioni, di scarso successo, dei loro desktop?
LT: Conosco persone con una padronanza avanzata dell'informatica che hanno deciso di rinunciare al desktop Linux solo perché sono rimasti frustrati dall'esperienza con Gnome e KDE, quindi hanno creato un precedente negativo. D'altra parte, a prescindere da loro non credo che le personi "normali", le nonne, la gente che non è appassionata di computer, avrebbe usato il desktop Linux. Mi spiace del precedente creato da Gnome e KDE ma, nel grande schema delle cose, secondo me è un dettaglio secondario.

LXP: Perché non usi il marchio Linux per creare un ambiente Linux di default?
LT: Non mi interessa. Non ho mai voluto occuparmi dell'aspetto tecnico. Preferisco continuare a lamentarmi della situazione, perché è catartico, e a discutere con la gente su Internet perché il mio passatempo preferito è discutere, non programmare. Allo stesso tempo voglio avere a che fare il meno possibile con il marchio Linux che è sempre stato un tormento, sin dall'inizio. I marchi registrati sono sempre un tormento. In verità la situazione legislativa per quanto riguarda i marchi registrati spinge le aziende a fare scelte stupide, perché i loro avvocati hanno la percezione che, se non facessero queste sciocchezze, perderebbero il controllo dei loro marchi. Sono fesserie, ma gli avvocati sono pagati per cercare il pelo nell'uovo. Di conseguenza, non vorrei comunque usare il marchio Linux.

LXP: Abbiamo letto che passi la maggior parte del tuo tempo a integrare codice nel kernel. Come fai a mantenere l'entusiasmo quando ormai non si programma più?
LT: L'operazione in sé è veramente velocissima nella maggior parte dei casi. Ci metto più tempo a copiare e incollare gli indirizzi Git e a leggere di che cosa si tratta...

LXP: Ed è sufficiente?
LT: È sufficiente per la maggior parte del codice. Non è la parte più coinvolgente, però. Quello che veramente mi fa infuriare è quando la gente fa cose stupide... il codice è importante ma nella realtà dei fatti il mio compito oggi non è di gestire la programmazione ma il flusso di lavoro delle persone. A volte succede che vengano fatte delle enormi fesserie e questa è la parte che tiene più acceso l'interesse... e poi posso urlare alla gente.

LXP: Cosa succede quando una società come Google lancia un mega progetto come Android, come hanno fatto quest'anno: ti avvisano prima?
LT: Mi aspettavo che ci fossero molti più problemi. Ne abbiamo discusso prima del lancio e la maggior parte dei problemi di Google Android non erano tanto legati a Google quanto allo sviluppo del kernel. Android sta andando molto bene, funziona e nessuno aveva un codice alternativo pronto che potesse essere compatibile con le esigenze di Google, perché avevano problematiche specifiche che nessun altro avrebbe risolto per loro.

LXP: Google ha fatto pressione per riunificare il kernel Android su cui stavano lavorando?
LT: Gli sviluppatori del kernel all'interno di Google odiavano il fatto di avere dei patch separati. In primo luogo perché non erano soddisfatti di essere fuori dal kernel e in secondo luogo perché dovevano fare del lavoro in più.

LXP: Quindi sono le persone che curano quei sottosistemi che si assumono effettivamente la responsabilità da parte di Google?
LT: Uno dei problemi di Android era la gestione dell'inattività del dispositivo. Vogliono che il sistema vada in modalità sleep molto aggressivamente ma che allo stesso tempo sia estremamente veloce a rispondere alle applicazioni che possono riattivarlo. Non funzionava bene con altre parti del codice per la gestione della carica e sotto certi aspetti non rispettava effettivamente la teoria del sistema. Aveva inoltre un impatto notevole su molti driver. C'era un effetto a cascata per questa interfaccia che serviva a Google e che avevano realizzato in un modo non previsto nel kernel standard e che influenzava i programmatori dei driver. A nessuno piace prendere ordini e a molti ha dato fastidio questa deviazione che creava problemi per una modifica marginale. D'altra parte, anche se non tutto ci entusiasma, alla gente in fondo non dispiace.





Mark Shuttleworth
fondatore di Ubuntu
Ottobre 2005

Linux Pro: Perché hai iniziato Ubuntu?
Mark Shuttleworth: Per una serie di motivi diversi. In primo luogo, fare qualcosa di utile: sono stato molto fortunato durante il boom delle dotcom [ha venduto la sua società Thawte a VeriSign per 575 miloni di dollari nel 1999] e in parte ho dovuto questa fortuna al fatto di essere stato esposto al software Open Source al momento giusto. Ho realizzato **Thawte** con Linux, **MySQL** e **Apache**. Avevo un programma filantropico molto sentito e in pratica ho bisogno di liberarmi di tutto ciò che ho acquisito. Ci terrei a farlo nel corso della mia vita quindi bisogna organizzarsi abbastanza in fretta. È una cosa a cui tengo molto dal punto di vista filantropico ed è anche un investimento per avere una parte nel determinare il futuro del mercato del software. Non vi posso dire come mi aspetto che si svilupperà il mercato del software perché il polverone delle mille battaglie impedisce di vederci chiaro...

LXP: Ci sono già molte distribuzioni Linux. Non sarebbe stato più pratico adattare alla tua visione una di esse?
MS: L'unica che davvero mi entusiasmava non si può modificare. In pratica quello che facciamo è il prodotto di questo mio pensiero. Ho considerato l'ipotesi di propormi come leader del progetto Debian ma ho pensato che c'è un altro modo per ottenere lo stesso effetto e che consiste nel creare qualcosa che traduce veramente in realtà questa visione. Rendere i nostri ideali disponibili liberamente e permettere alle altre persone di prendere ciò che preferiscono, le parti migliori per loro.

LXP: In effetti è molto popolare e sembrava essere uscito dal nulla...
MS: Beh, a diffenza di alcuni altri progetti nel settore, non siamo scesi a compromessi su alcuni degli attributi tecnici. Dò al mio team la massima libertà di fare le scelte tecniche che ritiene più opportune. Partire dalla base di Debian naturalmente è un aiuto grandissimo, quindi siamo stati fortunati e tutto ha funzionato al primo colpo. L'altra tipologia di persone che secondo me trova l'Open Source molto interessante è all'estremità opposta dello spettro degli utenti informatici. Sono persone che non sanno quasi nulla di computer e non hanno particolare interesse a imparare.

LXP: Alcuni pensano che Ubuntu abbia avuto successo a spese di Debian. Pensi che sia una critica corretta? Secondo te è importante?
MS: Certamente secondo me è importante sapere che alcune persone la pensano così, perché ci tengo che Debian abbia successo e che la gente abbia la percezione che ciò che stiamo facendo è costruttivo. Di conseguenza mi dispiace e mi preoccupa quando sento dire che Ubuntu ha penalizzato Debian. D'altra parte, sapevo già all'inizio del progetto che sarebbe stato controverso in seno alla comunità, perché spesso ci sono reazioni istintive che non esaminano bene i fatti prima di valutarli. Uno dei motivi per cui ho deciso di non sviluppare il mio progetto all'interno di Debian è il fatto che penso fermamente che una community Open Source possa fare qualsiasi cosa ma non possa fare tutto. Non c'è nessuna particolare montagna che un gruppo Open Source non possa riuscire a scalare, ma non possono essere in vetta a tutte le montagne contemporaneamente. La bellezza di avere un progetto Open Source è che possiamo mandare team diversi a scalare montagne differenti. Possiamo vincere entrambe le battaglie. Nella mia visione Ubuntu restringe le prospettive di Debian, perdendo qualcosa per ottenere più possibilità sotto altri aspetti.

LXP: Si può riversare interamente su un solo disco...
MS: Questa è un restrizione grandissima. Abbiamo sacrificato moltissimo per ottenere questo scopo. Se ti interessa particolarmente tutto ciò che non è su quel disco, non possiamo fare molto per aiutarti. L'altro lato della medaglia, però, è che otteniamo dei vantaggi da questa restrizione. Siamo nel ventunesimo secolo ed è ora di comportarsi di conseguenza. Dovremmo davvero capire come funzionano le organizzazioni distribuite. Ed è una sfida: è molto difficile costruire un'azienda in modo completamente distribuito. Dovremmo capire se è davvero possibile creare un'azienda con una visione puramente Open Source. Non so la risposta a questa domanda, ma penso che valga la pena di provarci.

LXP: Pensi che Linux debba conquistare i desktop per essere un successo?
MS: Linux è assolutamente pronto per alcuni desktop. Il punto è capire quali desktop. In base alle dimensioni della tua organizzazione, Linux è sempre utile per qualcosa. Non suggerirei però a un padre con tre figli di installare Linux sul suo computer, perché ne ha uno solo. »





Greg-Kroah Hartman
Maintainer del Kernel Linux
Luglio 2006

Linux Pro: Vari altri sistemi operativi usano driver Linux: Syllable, IBM K42...
Greg-Kroah Hartman: Hurd...

LXP: E Hurd, giusto! Secondo te è un'espressione dello spirito del code sharing o sul lungo termine avrà degli effetti negativi?
GKH: No, sono sorpreso che non condividiamo tutto. Le persone che lavorano su IBM K42 non vogliono scrivere il codice di un driver: vogliono concentrarsi sul loro kernel sperimentale. A nessuno piace scrivere i driver. Alcuni di noi lo apprezzano, ma molti ricercatori no, vogliono avere tutto operativo subito senza preoccuparsi di lavorare con i driver.

LXP: Perché a nessuno piace scrivere il codice dei driver? Sono difficili da programmare o testare?
GKH: Secondo me no. A me piace, è il mio lavoro. È una questione di approccio: tradizionalmente la gente pensa che i driver richiedano una programmazione di base, non interessante: sono il tipo di lavoro che si dà al nuovo arrivato in azienda. Spero che i tanti bravi programmatori che hanno lavorato con Linus abbiano cambiato questa percezione e i nostri driver sono noti per la loro qualità e stabilità complessiva. Il lavoro sulle reti è stato ottimo: SCSI e USB sono eccellenti e supportiamo nuovi dispositivi in un tempo più rapido di qualsiasi altro sistema operativo. Abbiamo fornito il supporto USB 2.0 per primi e supportato molte altre funzioni, tra cui il Bluetooth, prima degli altri sistemi operativi.

LXP: Ho dovuto installare Windows sul mio laptop. Non è facile.
GKH: Sì, offriamo il supporto per l'hardware più in fretta. Tutti gli sviluppatori di hardware usano Linux per i test. IA-64 è stato fatto su Linux, x86-64 è stato sviluppato su Linux. Lo possono fare, di conseguenza chi lavora sull'hardware ama Linux. Hanno il codice sorgente, così possono identificare i problemi del loro hardware... Il team di PowerPC ha fatto un ottimo lavoro. Hanno appena divulgato una pubblicazione su come installare Linux su PowerPC giganteschi multiprocessore senza firmware, senza BIOS. Non hanno dovuto aspettare il lavoro del team che si occupa del BIOS: se ne è occupato direttamente il gruppo dell'hardware.

LXP: Hai detto che al Kernel Summit del 2004 hai modificato un terzo del kernel. Ho fatto i conti: sono un milione e 200.000 linee di codice e 850.000 linee sono state rimosse, il che è straordinario. Mi sembra una revisione ciclopica.
GKH: Devi prendere questi numeri con le pinze. Sono le linee totali modificate: può capitare che si aggiungano e sostituiscano le stesse linee di codice, anche se di solito non è così. Stanno aggiungendo nuovi driver, rivedendo le API del kernel, migliorando vari aspetti.

LXP: Un aspetto su cui hai una posizione categorica sono i driver binari. Puoi spiegarci perché i driver binari, per esempio di Nvidia, sono illegali?
GKH: Il driver Nvidia di per se non è illegale. È molto semplice, basta parlare con un avvocato, qualifica che io non ho. La licenza GPL definisce esplicitamente il linking.

LXP: Unire codice GPL con codice non GPL?
GKH: Sì, quando unisci due codici, e hai bisogno di farlo quando carichi un modulo, quando unisci del codice nel kernel ottieni un'immagine di sistema che è coperta dalla licenza GPL. Non ci sono dubbi.

LXP: Come hanno fatto alcuni driver ad aggirare questa norma?
GKH: Puoi fare cose illegali se nessuno ti vede.

LXP: Certo, ma prima hai detto che il driver Nvidia non è illegale di per sè.
GKH: Perché non include niente di illegale. È l'utente a dover fare la compilazione e le unioni. L'utente però non può passare l'oggetto compilato ad altri senza violare la licenza GPL.

LXP: Vi facilita l'esistenza?
GKH: No, non ce la facilita affatto. I driver binari ci rendono la vita difficile. Gli utenti ci scrivono che hanno problemi con il loro kernel e se hanno un driver binario installato non sappiamo quale possa essere la causa delle loro difficoltà: il driver potrebbe sovrascrivere una qualsiasi parte del kernel e farlo andare in crash o bloccare delle funzioni e noi non possiamo saperlo. Il punto è che se ci notifichi che il kernel si blocca e hai installato un driver binario la nostra risposta è sostanzialmente che non possiamo supportarlo... se hai un problema, devi risolverlo da solo. Ci capita che la gente, prima di mandarci il messaggio a schermo che gli compare quando il kernel si blocca, lo modifichi in modo che sembri che non ci siano driver binari installati, perché sanno che in quel caso non c'è supporto.

LXP: Durante la causa legale tra SCO e IBM per Linux non avete avuto un periodo di dubbio passato a spulciare il codice per vedere che non ci fosse davvero qualcosa di proprietà di SCO?
GKH: No. Tutto il codice di Linux è ben documentato e sappiamo esattamente da dove viene. Si potrebbe invece girare la domanda: da dove viene il codice dei sistemi operativi proprietari? Non si può sapere se c'è codice nostro... LXP



CALENDARI-AGENDA 2016
IN ANTEPRIMA SOLO PER TE!

Prenotali subito!

REGALALO A UN AMICO!
Glielo spediremo a tuo nome e ti penserà tutto l'anno



Sarai il primo a riceverli perché te li spediremo appena stampati!

Scegli i tuoi calendari preferiti

Ordinali subito su www.lisagoodmancalendar.it oppure utilizzando questo coupon

	Q.TÀ	PREZZO	SUBTOTALE
CALENDARIO-AGENDA DELLA FAMIGLIA F.TO CM30X42,5		€ 6,00	
CALENDARIO-AGENDA DEI NONNI F.TO CM15X50		€ 5,90	
CALENDARIO-AGENDA PER LEI F.TO CM15X50		€ 5,90	
SPESE DI SPEDIZIONE DA AGGIUNGERE AL TOTALE			€ 3,90
Totale Ordine			

Se approfitti dell'offerta 3x2 alla riga q.tà metti "0" al calendario meno costoso

SE VUOI ORDINARE VIA POSTA O VIA FAX, COMPILA QUESTO COUPON

Ritaglia o fotocopia il coupon, invialo in busta chiusa a:
Sprea S.p.A. socio unico Sprea Holding S.p.A. Via Torino, 51 20063 Cernusco s/n (MI), insieme a una copia della ricevuta di versamento. Oppure via fax al numero 02.700537672
Per ulteriori informazioni puoi scrivere a info@calendarioagenda.it o telefonare al 02.87158224.

NOME
COGNOME
VIA
N° C.A.P. PROV.
CITTÀ
TEL.
E-MAIL
scrivi qui il tuo indirizzo mail ti avvertiremo dell'avvenuta spedizione

Voglio regalare questo calendario a:

NOME
COGNOME
VIA
N° C.A.P. PROV.
CITTÀ

SCELGO IL SEGUENTE METODO DI PAGAMENTO E ALLEGO:

Indica con una ✔ la forma di pagamento desiderata

- ☐ **Ricevuta di versamento su CCP 99075871**
  o bonifico bancario sul conto IBAN IT 05 F 07601 01600 000099075871
  intestato a Sprea S.P.A. Via Torino 51 - 20063 Cernusco Sul Naviglio MI
  - Ricorda di segnalare nella CAUSALE il nome del soggetto scelto -
- ☐ **Carta di Credito**

N.
(Per favore riportare il numero della Carta indicandone tutte le cifre)

Scad. CVV *(Codice di tre cifre che appare sul retro della carta di credito)*

Nome e Cognome del Titolare

Data________________ Firma del titolare____________________

CartaSi Diners Club International VISA VISA Electron MasterCard postepay PayPal

PRENOTALI SUBITO SU www.lisagoodmancalendar.it



Premiata Amministeria Dott. Brown

Jolyon Brown
Quando non fa consulenze su Linux, Jolyon passa il suo tempo lavorando alla sua grande ambizione: trovare un motivo per usare Emacs...

Tecniche esoteriche per i sysadmin dai recessi più impenetrabili della sala server

Differenze di genere

Recentemente ho preso parte a un piccolo evento dedicato al **DevOps** amichevole, interessante e seguito quasi unicamente da uomini. Erano presenti solo tre donne all'interno di un pubblico di circa 60 persone. Non preoccupatevi, non ho intenzione di scrivere uno di quegli articoli in cui un uomo pontifica sul perché "non ci sono donne nell'IT". Rimane però il fatto che in particolare l'ambiente DevOps/Sysadmin sembra soffrire di uno sbilanciamento di genere, nonostante i molti notevoli sforzi (compresi quei movimenti e quegli eventi pensati specificamente per le donne e per altri segmenti poco rappresentati della società) volti a incrementare la presenza femminile nell'IT. Non mi sembra giusto che le donne (o qualsiasi altro gruppo poco rappresentato) non intraprendano carriere che risultano sempre più flessibili e relativamente ben pagate. Ho passato ad alcune mie conoscenze femminili e ad alcune ex colleghe una bozza di questo articolo e ho ricevuto in cambio alcune buone idee e, devo dire, anche qualche critica costruttiva su come collaborare per migliorare la situazione. Il suggerimento più comune era quello di fornire una guida e dei buoni consigli per iniziare a lavorare nell'IT. Mi sono perciò offerto come volontario per fornire, attraverso vari canali, una guida alle donne che volessero lavorare nel settore del DevOps/Sysadmin, e sono felice di poter dire che ho già ricevuto degli elogi per questo! È anche importante parlare chiaro. È vero che il sessismo becero non è più tollerato nei luoghi di lavoro, ma mi è capitato di lavorare in aziende in cui le donne erano sottilmente meno rispettate degli uomini che svolgevano lo stesso lavoro. Anche la formazione è di vitale importanza. Ho passato un po' di tempo su **https://adainitiative.org/**, un'iniziativa sponsorizzata da Google, Red Hat e dalla Linux Foundation che mette a disposizione parecchio materiale sull'argomento.

Contenitori & C.

Le nuove release si susseguono mentre i venditori di sistemi di gestione delle configurazioni e PaaS si danno da fare per incrementare il mercato DevOps

Gartner, in una sua recente analisi, afferma che il mercato degli strumenti per DevOps raggiungerà i 2,3 miliardi di dollari nel 2015 e che il settore si "evolverà in una strategia *mainstream*" entro il prossimo anno. Non c'è quindi da meravigliarsi se nei mesi scorsi si sono susseguite novità, conferenze stampa e nuove release di prodotti per il DevOps, il cloud e i contenitori. PuppetLabs, Chef, Pivotal e persino Microsoft hanno annunciato nuovi prodotti, lanciato progetti o rilasciato aggiornamenti di software esistente. Sia PuppetLabs sia Chef (che ha tenuto la sua conferenza annuale ChefConf all'inizio di aprile) hanno aggiornato la loro offerta, focalizzata sulla gestione automatica, sul *provisioning* e sul *testing*. I contenitori e la nuvola sono stati i veri protagonisti di entrambi gli annunci, mentre il supporto per **Docker** è diventata una caratteristica irrinunciabile di tutti i contendenti sul mercato. Pivotal ha lanciato ufficialmente il suo prodotto Open Source **Lattice**, pensato per aiutare nel *deployment* e nella gestione di contenitori Docker nella (avete indovinato) nuvola. Usando un sottoinsieme del loro software **Platform as a Service** (CloudFoundry), Lattice offre la schedulazione automatica, il bilanciamento del carico e la gestione dello stato di salute delle istanze e dei log. Il software può essere scaricato da GitHub (e molto probabilmente sarà il protagonista di un prossimo articolo dell'Amministreria). **CoreOS** (vedere le pagine seguenti) ha annunciato una piattaforma commerciale basata su Kubernetes, **Tectonic**, che offre supporto per, aspettate..., i contenitori Linux, il deployment automatico, *workflow* e *dashboard*. Anche Microsoft è scesa nell'arena annunciando che i contenitori Hyper-V saranno un'opzione per il futuro Windows 2016 e che, cosa davvero notevole, saranno gestibili da Docker, confermando così la recente tendenza dell'azienda verso una maggiore apertura nei confronti dell'Open Source. L'ultima release di **OpenStack**, nome in codice Kilo, è stata rilasciata il 30 aprile. Purtroppo però ci sono alcune cattive notizie. HP ha annunciato che chiuderà il suo servizio cloud basato su OpenStack a causa della concorrenza di Amazon e Microsoft. Allo stesso modo Nebula, una delle prime aziende ad adottare OpenStack con lo scopo di facilitare l'adozione di piattaforme cloud Open Source, ha chiuso i battenti in aprile.

CoreOS: l'infrastruttura del futuro

Promette "capacità computazionale scalabile e gestibile dinamicamente" stile Google: sarà vero?

Per me la più grande attrattiva dell'Open Source e di Linux in particolare consiste nel modo con cui incoraggia la nascita di nuove idee. Qualcuno, da qualche parte, se ne viene fuori con una maniera per risolvere un particolare problema, lo implementa, rende Open Source il risultato e da quel momento l'idea prende una vita sua propria. Potrebbe fiorire, seccarsi o languire, oppure venire inglobata o ispirare un po' di altre idee. Non credo di esagerare se affermo che il modo in cui le idee si evolvono, usando Internet come strumento di diffusione, è uno degli aspetti più sorprendenti della tecnologia odierna. Ovviamente il rovescio della medaglia di questa esplosione di creatività è che c'è una lista pressoché infinita di cose Open Source da vedere ed esaminare. Io ho un'enorme lista di cose da provare e per un bel po' di tempo CoreOS ne ha fatto parte, senza peraltro arrivare mai in cima alla lista. Ciò è principalmente dovuto al fatto che non ho nessun cliente fortemente interessato a usarlo (il progetto è nato nel 2013) e quindi ha dovuto aspettare il suo turno. Ma ora ho trovato una scusa per farlo. Ora che il momento è arrivato, mi chiedo come mai ci abbia messo tanto. Il sito Web di CoreOS descrive un sistema che si propone di risolvere parecchi dei miei problemi di DevOps/amministrazione di sistema, tra cui l'automazione, la facilità nel deployment di applicazioni, la sicurezza, l'affidabilità e la scalabilità. Quando devo provare qualcosa di nuovo, come in questo caso, di solito parto da uno scenario che potrebbe presentarsi con un potenziale cliente, in modo da verificare come potrebbe adattarsi a un mio probabile caso d'uso. Perciò in questo articolo e nel successivo valuterò CoreOS come possibile soluzione per un cliente che voglia migrare i suoi siti Web, ospitati da un servizio esterno, su CoreOS. Supponiamo che il cliente stia cercando una maniera per ridurre i costi della sua infrastruttura e pensi che le funzionalità offerte da CoreOS possano essere la maniera giusta per farlo. Cominciamo quindi con l'esaminarle per vedere se sono un buon punto di partenza.

Partiamo dal centro

CoreOS si basa su **ChromeOS/Chromium**, il sistema operativo per browser di Google, che, ovviamente, contiene al suo interno il kernel Linux (la versione attuale si basa su Gentoo). L'attenzione non è però rivolta a fornire le funzionalità presenti in una normale distribuzione desktop; CoreOS è pensato unicamente per la gestione e il deployment di applicazioni, cercando di minimizzare tutto quello che di solito ha a che fare con questo genere di cose, come l'applicazione di patch, la manutenzione, lo spostamento di ambienti di lavoro tra sistemi offline, ecc. La prima cosa che colpisce un decrepito amministratore di sistema come me è il fatto che CoreOS non usa un tradizionale sistema per la gestione degli aggiornamenti come **yum** o **apt** (proprio adesso che finalmente ero riuscito a capire tutte le opzioni di questi comandi). CoreOS offre invece aggiornamenti automatici, scaricati direttamente dal sito del progetto, più o meno allo stesso modo in cui lavora il browser **Chrome** di Google. In secondo luogo, al centro di CoreOS c'è un host di controllo "leggero". Le app girano su di esso in una maniera simile a un pseudo *hypervisor*, ma al posto di macchine virtuali CoreOS usa **Docker**, che fa girare le applicazioni all'interno di contenitori. Oltre a questo CoreOS usa **Fleet**, uno strumento che permette di distribuire i contenitori su un cluster di macchine ed è in grado di gestire regole di affinità (cioè "Non far girare tutti i componenti della mia applicazione ad alta disponibilità sullo stesso host fisico"). Infine c'è **etcd**, un programma per la scoperta dei servizi, che viene distribuito insieme al sistema operativo. Questo vi permette di conservare le configurazioni al di fuori delle applicazioni: non è più necessario definire all'interno delle applicazioni le stringhe per la connessione al database e cose simili. Inoltre è in grado di replicare il tutto sui vari nodi di un cluster. Interessante, vero? Procediamo con i test.

Configurare un sistema di test

CoreOS è in grado di lavorare con ampia gamma di ambienti per il deployment. È stato fatto un grosso sforzo per assicurarsi che possa essere utilizzato con i protagonisti del mercato cloud (Amazon, Rackspace, Azure, Google Compute Engine, Digital Ocean) e che anche **OpenStack** e **Vagrant** possano essere utilizzati. Dato che sto simulando come potrebbe essere valutato da un cliente, userò Vagrant per »

CoreOS + Kubernetes = Tectonic (ma non Open Source)

Appena prima di andare in stampa CoreOS ha annunciato un nuovo investimento da parte di Google Ventures e il lancio di un nuovo servizio: **Tectonic**. Si tratta di un'offerta commerciale (CoreOS offre servizi di supporto a pagamento, analogamente a quanto fanno Red Hat e altri) e contiene **Kubernetes**, un progetto Open Source di Google. Kubernetes è un sistema di orchestrazione per contenitori **Docker** che li distribuisce sui nodi di un cluster e li gestisce attivamente. La cosa interessante per i geek appassionati di infrastrutture è che si basa sul codice interno usato dall'infrastruttura di Google (che gestisce servizi come Search e Gmail). Viste le capacità di elaborazione del gigante dei motori di ricerca (oltre all'impatto sull'IT degli articoli dei suoi ricercatori) e la misteriosa aura che circonda i suoi enormi data center, qualsiasi possibilità di dare un'occhiata al suo funzionamento interno risulta estremamente interessante. CoreOS ha però deciso di non rendere Open Source la sua implementazione, cosa abbastanza spiacevole, anche se l'azienda ha ribadito che tutti gli altri componenti di CoreOS resteranno Open Source. Per sottolineare questo Tectonic avrà un marchio suo, separato da CoreOS. Kubernetes non è comunque l'unica soluzione disponibile. **Apache Mesos** è un *framework* simile, mentre Docker ha ispirato un certo numero di progetti PaaS (*Platform as a Service*), tra cui Deis. Dato poi che Kubernetes è Open Source, sono sicuro che vedremo nascere altre alternative, perché l'Open Source tende alla fine a prevalere.

» l'installazione locale su un desktop Linux. La documentazione di CoreOS raccomanda di usare una versione di Vagrant superiore alla 1.6.3, quindi userò i file Deb scaricati direttamente da **http:://www.vagrantup.com** per installarlo sul mio sistema Ubuntu 14.04, su cui sono già presenti per default versioni precedenti. Ho anche installato **Virtualbox** per essere in grado di far girare macchine virtuali. A questo punto non mi rimane che clonare il repo Git di CoreOS per Vagrant:

```
git clone https://github.com/coreos/coreos-vagrant.git
cd coreos-vagrant
```

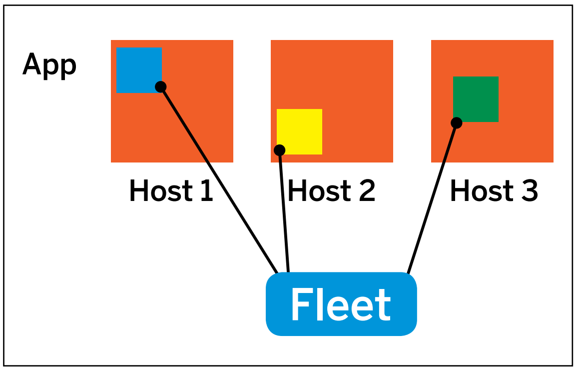
Guardando nella directory **coreos-vagrant** appena creata noto due file **.sample**. Il primo da esaminare è **user-data.sample** (e la prima cosa da fare è rinominarlo in **user-data**). Il file contiene la seguente riga commentata:

```
#discovery: https://discovery.etcd.io/<token>
```

Per il test devo procurarmi un token di esempio dal servizio pubblico di **discovery** di CoreOS (si tratta di un'istanza di **etcd** accessibile da Internet), raggiungibile all'URL **https://discovery.etcd.io/new**. Accedendovi con un browser (o con **curl**) si ottiene una stringa con un token alla fine, che va copiata nel file **user-data**, eliminando il segno di commento all'inizio della riga:

```
discovery: https://discovery.etcd.io/
d72db3274bbfcff3069683ba649e6c69
```

Questo servizio pubblico di **discovery** può essere usato per



› ***Fleet* è in grado di distribuire i contenitori all'interno di un cluster CentOS. Può applicare regole di affinità, spostare i contenitori durante la manutenzione e garantire che il numero corretto di istanze sia attivo e in esecuzione**

evitare di configurarne uno in proprio, ma vale la pena di notare che molti client non sono in grado di accedervi o (forse per una scelta deliberata) non vogliono utilizzare un servizio offerto da terze parti. La configurazione di un servizio di questo genere verrà descritta in un futuro articolo. Fatto questo procedo a rinominare **config.rb.sample** in **config.rb**, poi lo apro e modifico le seguenti righe:

```
#$num_instances=1
...
#$update_channel='alpha'
$num_instances=3
...
$update_channel='stable'
```

Ho portato il numero di istanze che voglio in esecuzione a tre perché poi voglio provare il clustering e Fleet. Dato che sono nella fase iniziale del test di CoreOS, ho scelto di usare la release stabile piuttosto che correre il rischio di passare ore alla caccia di bug nella release alpha della prossima versione, di cui non so nulla. Se, dopo aver compreso quelle disponibili nella release stabile, scoprirò che la nuova release mette a disposizione nuove funzionalità di cui non posso fare a meno la installerò. È giunto il momento di provare a far partire il sistema: basta dare il comando

```
vagrant up
```

La prima volta che l'ho lanciato Vagrant ha scaricato per me un'immagine di CoreOS per VirtualBox e ha fatto partire il tutto. Dopo alcuni minuti ho potuto dare il comando

```
vagrant status
```

che ha prodotto il seguente output:

```
Current machine states:
core-01        running (virtualbox)
core-02        running (virtualbox)
core-03        running (virtualbox)
```

Ho quindi potuto proseguire con il comando

```
vagrant ssh core-01
```

che mi ha permesso di accedere alla prima VM. Un veloce sguardo in giro ha confermato che si tratta di un sistema Linux, anche se ben poco popolato.

etcd: il sistema di configurazione e scoperta

Secondo il sito di CoreOS "**etcd** è un archivio distribuito Open Source di coppie chiave valore con funzionalità di configurazione condivisa e scoperta di servizi per i cluster CoreOS". Cosa significa esattamente? Le applicazioni (all'interno dei contenitori) che girano su CoreOS possono leggere e scrivere dati in etcd. I dettagli per la connessione a un database sono un caso d'uso comune per questo genere di dati. Per fare questo è disponibile una semplice API HTTP e c'è anche un comodo comando (**etcdctl**). Nel sistema di test usato per questo articolo diamo:

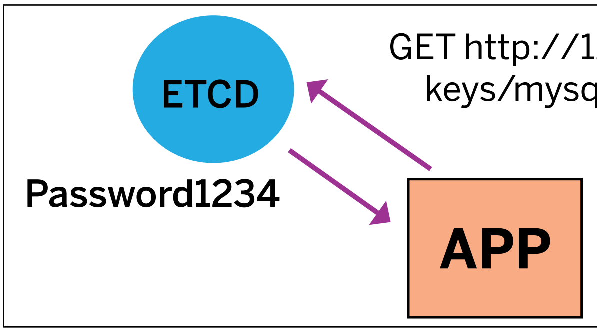
```
etcdctl set /lxf hello
```

in una VM e poi su un'altra

```
etcdctl get /lxf
```

per recuperare la stringa "hello". Questo significa che possiamo cercare i dati da qualsiasi nodo del cluster e che i dati risulteranno sempre consistenti. I servizi possono essere scoperti così, basta che quando partono registrino in etcd le informazioni necessarie per utilizzarli.

Gli altri servizi possono vedere se ci sono degli aggiornamenti e modificare di conseguenza il loro comportamento. Per esempio un bilanciatore di carico potrebbe scoprire dei nuovi servizi di back-end senza intervento da parte dell'amministratore di sistema. etcd è altamente disponibile: è in grado di gestire la caduta di più nodi. Quando un cluster diventa grande occorre prendere qualche decisione architetturale, per esempio sull'utilizzo di proxy per la gestione del traffico. Come c'è da aspettarsi, etcd non è l'unico progetto Open Source in grado di fare cose di questo genere. **Zookeeper**, **Consul** e **Doozerd** sono altri esempi. Ognuno ha i suoi punti di forza e le sue debolezze. In quest'area dello sviluppo di infrastrutture è in atto una rapida crescita.



› **Servizi e applicazioni possono recuperare dati da *etcd* e avere risposte consistenti all'interno di tutto il cluster CoreOS**

Il comando **ps** mostra in esecuzione **Systemd** e il demone **Docker**, oltre a qualcosa chiamato **update_engine**, ma poco altro.

Aggiornamenti automatici

Fin qui tutto bene. La prima delle cose che voglio verificare è l'efficacia del sistema di aggiornamento usato da CoreOS, dato che sarà una delle cose più importanti per i miei clienti. Il file **/etc/motd** delle istanze Vagrant mostra che stanno eseguendo CoreOS versione 607.0.0 (la più recente stabile secondo il sito **http://coreos.com**). CoreOS gestisce gli aggiornamenti per mezzo di due partizioni radice note come **USR-A** e **USR-B**. Subito dopo l'installazione il sistema si avvia usando la prima delle due partizioni, che viene montata in sola lettura su **/usr**. Il motore degli aggiornamenti installa una nuova copia completa del sistema operativo sul filesystem alternativo non montato su **/usr** (che inizialmente è vuoto per risparmiare spazio). Quando il sistema si riavvierà verrà usata la versione appena aggiornata del sistema operativo. In caso di errore è sempre possibile farlo ripartire usando la versione precedente. Per vedere le partizioni sul sistema ho usato il comando **cgpt**, che ha prodotto un sacco di informazioni (ridotte qui di sotto per problemi di spazio):

```
sudo cgpt show /dev/sda

270336 2097152 3 Label: "USR-A"
  Type: Alias for coreos-rootfs
  UUID: 7130C94A-213A-4E5A-8E26-6CCE9662F132
  Attr: priority=1 tries=0 successful=1
2367488 2097152 4 Label: "USR-B"
  Type: Alias for coreos-rootfs
  UUID: E03DD35C-7C2D-4A47-B3FE-27F15780A57C
  Attr: priority=0 tries=0 successful=0
```

Sotto **/usr/boot/grub** ho trovato nel file **menu.lst** le righe corrispondenti a **USR-A** e **USR-B**. Mi sembra una soluzione piuttosto elegante per gli aggiornamenti, a patto di essere in grado di sapere cosa sta facendo il motore di aggiornamento. Dato che si tratta di un sistema operativo basato su Systemd, posso usare il comando **journalctl** per farlo:

```
journalctl -f -u update-engine
```

Ho così scoperto alcune righe nel registro del sistema che dicevano che il sistema operativo non era in grado di risolvere l'host **public-update.core-os.net**: un problema della mia connessione. Ho anche scoperto che il processo di aggiornamento si basa sul progetto Omaha di Google.

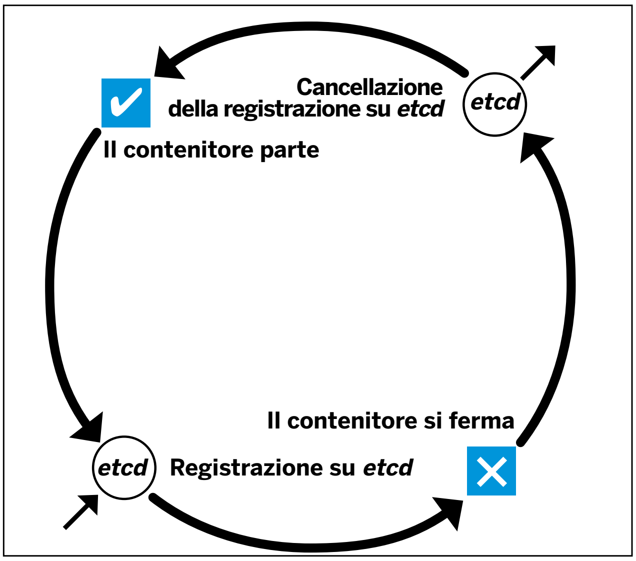
Fleet

Ora voglio occuparmi di **Fleet**, il gestore di cluster. È in grado di manipolare unità di **Systemd** (gli altri elementi di Systemd sono noti come bersagli, *target* in inglese). Queste unità sono file di configurazione che descrivono le proprietà di un processo; nel mio caso avranno più che altro a che fare con Docker. In CoreOS i file unità di Systemd si trovano sotto **/etc/systemd/system**. Se voglio vedere lo stato del mio cluster Fleet posso usare il comando:

```
fleetctl list-machines
```

Nel mio caso mostra tre macchine Vagrant:

```
MACHINE         IP            METADATA
c9fc38db...     172.17.8.101     -
ca1115a3...     172.17.8.103     -
e7713580...     172.17.8.102     -
```



› **Quando un contenitore parte o si ferma aggiorna il suo stato registrato in *etcd* in modo che gli altri sistemi possano eseguire le opportune azioni (pensate a un bilanciatore di carico che scopre che alcuni servizi di back-end vanno on/offline)**

Nel corso del test ho avuto dei problemi che ho scoperto essere dovuti a errori nel file **user-data**. Una volta l'ho rinominato per errore **user.data**. Un'altra volta ho sovrascritto la parola chiave "discovery", cosa che ha dato luogo a errori di **Yaml**. Un paio di comandi aiutano a risolvere questo genere di problemi:

```
systemctl status -l fleet
journalctl -b -u fleet
```

A questo punto sul mio sistema è tutto OK, quindi voglio avviare un servizio di test. Sul sito di Docker c'è un contenitore di test di esempio che ho adattato per il mio scopo. Ho creato il seguente file, chiamato **testapp.service**, sulla mia prima VM **Vagrant**:

```
[Unit]
Description=TestApp
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill testubuntu
ExecStartPre=-/usr/bin/docker rm testubuntu
ExecStartPre=/usr/bin/docker pull ubuntu
ExecStart=/usr/bin/docker run --name testubuntu ubuntu /
bin/sh -c "while true; do echo Hello Linux Format; sleep 1;
done"
ExecStop=/usr/bin/docker stop testubuntu
```

Posso avviare il servizio e tenerlo sotto controllo con i comandi **fleetctl**:

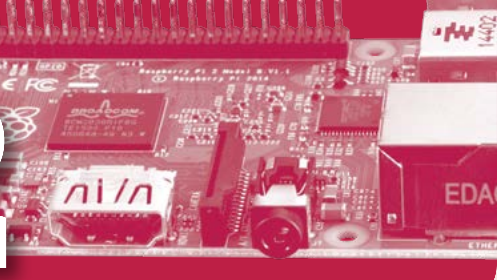
```
fleetctl start testapp.service
leetctl list-units
```

e ho così la soddisfazione di constatare che il servizio è in esecuzione!

```
core@core-01 ~ $ f
UNIT            MACHINE                    ACTIVE   SUB
testapp.service c9fc38db.../172.17.8.101   active   running
```

Piuttosto semplice, non trovate? Nel prossimo articolo esaminerò in maggior dettaglio sia **Fleet** sia **Docker** all'interno di CoreOS e proverò a realizzare quello che i miei clienti mi chiedono: un servizio di hosting resiliente. LXF

Pi & Pi 2



Tutte le news, le recensioni e i tutorial sul Raspberry Pi



BEN NUTTALL **è l'Education Developer Advocate alla Pi Foundation.**

Benvenuti...

Al nostro ufficio di **Cambridge della** Pi Foundation, chiamato Pi Towers dagli amici, ci siamo dati da fare con diversi progetti di hardware, software e didattica. Come forse avrete sentito, stiamo per mandare due Pi sulla Stazione Spaziale Internazionale (ISS) alla fine dell'anno per svolgere delle attività di esplorazione in collaborazione con le scuole, come parte della missione Principia dell'astronauta britannico dell'ESA Tim Peake. Mentre abbiamo completato la API Python per la scheda sensore e abbiamo scritto i materiali didattici, la Pi Model B+ Pi e il suo SenseHAT sono in corso di verifica con test di sicurezza per ottenere la certificazione al volo spaziale e siamo in trepida attesa di avere l'OK finale dall'ESA. Tim Peake tornerà sulla Terra dopo sei mesi di missione ma le Pi resteranno nello spazio. Non abbiamo ancora fatto piani per il loro futuro ma aspettiamo di scoprire cosa hanno in serbo il 2016 e il 2017. Abbiamo creato un allestimento alla Foundation of Art and Creative Technology (FACT) a Liverpool come parte della mostra Build Your Own. Si chiama **Neurotic Machines** (Macchine Nevrotiche) ed è figlia dell'immaginazione di Rachel Rayns, Produttore Creativo della Fondazione. Comprende un giardino verticale, una serie di vasi colorati e un pannello di controllo pieno di levette e potenziometri analogici, un pulsante gigante e tutto ciò che serve per gestire un giardino con un robot. Ci siamo divertiti molto a costruirlo e ha un aspetto fantastico. Per saperne di più: **http://bit.ly/LXFneuroticPI**.

Picademy USA

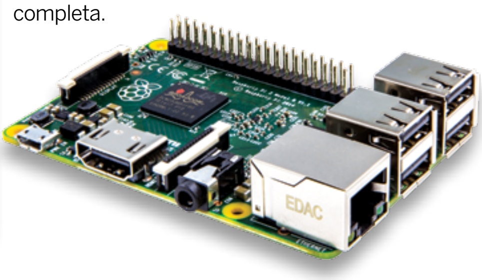
La fondazione annuncia che **Picademy** partirà alla conquista del Nuovo Mondo!

Picademy, il programma didattico professionale di Raspberry Pi, ha avuto un grande successo nel Regno Unito, dove è stato seguiro da circa 200 insegnanti in soli otto eventi. La Foundation vuole insegnare a centinaia di migliaia di educatori e per ottenere questo tipo di risultato il programma si è esteso a includere gli USA. Raspberry Pi Foundation ha raggiunto accordo con il Computer History Museum di Mountain View in California (**www.computerhistory.org**) per lanciare una serie di corsi pilota Picademy USA all'inizio del 2016. Il primo corso negli Stati Uniti durerà due giorni, dando a 25 insegnanti l'opportunità di vedere come si può usare la Raspberry Pi in classe. Alla fine del corso gli insegnanti diventeranno Raspberry Pi Certified Educators, ossia Educatori Certificati Raspberry Pi. I primi corsi si terranno in diversi Stati degli USA con lo scopo di formare circa 100 insegnanti. Se siete interessati a scoprire di più su Picademy oltreoceano, potete visitare l'indirizzo **https://www.raspberrypi.org/picademy/usa**. Nel frattempo Picademy nel Regno Unito continua ad andare benissimo e ad accogliere educatori di tutti i Paesi, quindi se siete insegnanti, capite l'inglese e siete interessati a sapere come partecipare andate a vedere il sito ufficiale Raspberry Pi (**www.raspberrypi.org**).

La Pi è prima... e anche terza!

La Pi è sempre più amata

Due popolari siti internazionali, **LinuxGizmos.com** e **Linux.com,** hanno fatto un sondaggio congiunto per scoprire quali sono i tre computer a scheda singola e open. spec basati su Linux o Android. I 1.721 lettori che hanno risposto hanno decretato un chiaro vincitore: la **Raspberry Pi 2 Model B**. Se arrivare primi nelle scelte dei lettori ci ha fatto piacere, ancora più straordinario è il fatto che la Model B non fosse l'unica Pi nei primi 3 posti. Il secondo posto infatti è andato a **Beaglebone Black**, ma al terzo si è posizionata la **Raspberry Pi Model B+**. Date un'occhiata ai risultati del sondaggio su **http://bit.ly/HackerSBCSurvey2015** per vedere la classifica completa.



Chi vuole un palmare con la Pi?

Costruite il vosto handheld

Il **Mini Raspberry Pi Handheld Notebook** di Adafruit è un dispositivo palmare compatto. Ha il case stampato in 3D, uno schermo touchscreen da 3,5 pollici e una batteria da 2.000 mAh. Se non volete lo schermo sporco di ditate, per controllare il dispositivo ci sono anche una mini tastiera chiclet e un trackpad. Adafruit suggerisce di usarlo per controllare Webcam, mandare **http://bit.ly/HackerSBCSurvey2015** musica in streaming e giocare con avventure testuali oltre che per mille altri scopi. Per ulteriori dettagli e per scoprire come costruirvi un palmare voi stessi, scaricando immagini del software e schemi, visitate il sito: **http://bit.ly/RaspPiHandHeldNotebook**.



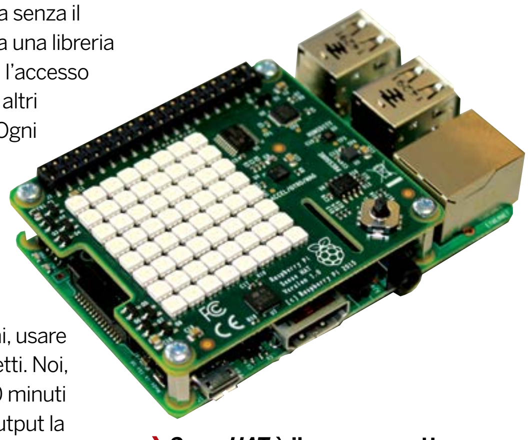
AstroPi SenseHAT

Spazio, l'ultima frontiera. Grazie a questo progetto la Raspberry Pi è pronta a solcare i venti spaziali allo scopo d'insegnare la programmazione

In breve

» La più recente tra le schede d'espansione create dalla Raspberry Pi Foundation. Integra tantissimi sensori utili per diversi esperimenti scientifici. La scheda fa parte di un progetto che vedrà la creazione di esperienze di laboratorio da parte di studenti delle scuole inglesi che verranno poi ripetute sulla ISS alla fine di quest'anno.

Da quando è nata, la Raspberry Pi è stata protagonista di svariate avventure che l'hanno portata in alto mare, all'interno di robot e ora anche nello spazio, grazie al progetto **AstroPi**. Si tratta di un progetto scientifico avviato in Inghilterra che ha visto coinvolte diverse scuole: gli studenti devono inventare degli esperimenti che, se selezionati, saranno eseguiti sulla Stazione Spaziale Internazionale (ISS) dall'astronauta Tim Peake. AstroPi è anche il nome di questa scheda d'espansione, collegata al progetto e progettata per le RP A+, B+ e la 2 (anche se può funzionare con tutti i modelli). Le dimensioni di AstroPi fanno sì che sia grande quanto la RP A+ e che si possa posizionare alla perfezione sulla B+. Questa scheda è una piattaforma per la cattura di dati acquisibili grazie al gran numero di sensori presenti. Per iniziare, trovate un sensore per l'umidità e la temperatura, al suo fianco invece ce n'è uno che misura la pressione. Infine è presente un elemento che combina giroscopio, accelerometro e magnetometro per i dati di posizione. I progettisti della scheda, non contenti, hanno aggiunto anche una griglia di 8x8 LED multicolore (neopixel) e un piccolo joystick.
La matrice di LED è molto luminosa (consigliamo l'uso di un diffusore, come un pezzo di carta, per proteggere la vista). Ma l'hardware è nulla senza il software, quindi AstroPi usa una libreria per Python 3 che consente l'accesso immediato ai sensori e agli altri componenti della scheda. Ogni elemento ha una funzione corrispondente, all'interno della libreria, in grado di gestire il flusso di dati generato. In questo modo è facilissimo, anche per i programmatori più giovani, usare la scheda per i propri progetti. Noi, per esempio, in meno di 20 minuti siamo riusciti a inviare in output la lettura della temperatura sulla matrice di LED controllandone il colore in base ai valori registrati. AstroPi ha un campo d'impiego molto vasto: dalla semplice registrazione della temperatura nel tempo a complessi esperimenti scientifici per il monitoraggio della pressione dell'aria ad alta quota usando palloni aerostatici o droni. I più fantasiosi, potrebbero, invece collegare AstroPi e MinecraftPi usando i sensori della board per controllare il mondo di gioco e la sua matrice di LED per indirizzare il giocatore verso posizioni nascoste.

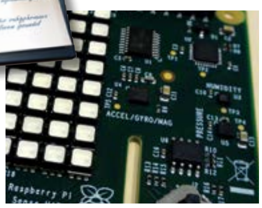


› ***SenseHAT* è il nome corretto dell'AstroPi. È progettata per funzionare con i modelli A+, B+ e 2 della piccola Raspberry Pi**

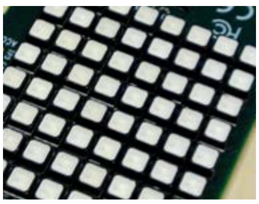
Portami sulla Luna

Come detto prima, la destinazione programmata per l'AstroPi è la ISS, dove si terranno una serie di esperimenti creati dagli studenti. Il codice e i dati generati da questi esperimenti verranno poi liberamente condivisi, in modo che chiunque possa ripetere le operazioni e confrontare i risultati con quelli ottenuti nello spazio. È la prima volta che si tenta di realizzare un progetto del genere, e non vediamo l'ora di vederne i risultati. Grazie ai vari sensori dell'AstroPi, come quelli per la pressione e la temperatura, potremo scoprire come fluttuano questi fattori all'interno della Stazione Spaziale Internazionale mentre orbita attorno al nostro pianeta. La Raspberry Pi Foundation, Airbus e l'ESA (*European Space Agency*) hanno lavorato assieme per circa un anno per progettare, controllare e certificare questa scheda d'espansione in modo che fosse pronta al viaggio nello spazio. Verrà usata assieme al modello B+ della RP: è stata fatta questa scelta perché ci vuole un po' di tempo a certificare la scheda e la versione 2 è troppo recente. La ISS ha delle misure di sicurezza molto strette e la RP non potrà essere alimentata dai circuiti della stazione spaziale, ma verrà usata la porta USB di un portatile. Insomma, AstroPi è davvero una fantastica scheda d'espansione per chi vuole cimentarsi nei più svariati esperimenti scientifici. LXP

Caratteristiche



Sensori
Alcuni dei sensori presenti gestiscono più di una funzione, come quello che misura temperatura e umidità.



Matrice di LED
I 64 LED di cui è composta la matrice possono essere usati per visualizzare i dati registrati dai sensori usando forme e colori.

LINUX PRO Giudizio

AstroPi SenseHAT

Sviluppatore: Raspberry Pi Foundation
Web: www.raspberrypi.org
Prezzo: circa 35 $ (31,20 €)

Caratteristiche	9/10
Prestazioni	9/10
Facilità d'uso	9/10
Qualità/Prezzo	9/10

» *La sua facilità d'uso, assieme ai tanti sensori presenti, la rendono utilissima per gli amanti della scienza.*

Il voto di Linux Pro **9**

Pigame: creare un gioco... sonoro

Linux Pro vi sfida a mettervi alla prova contro la RP... ma fate attenzione: quando sentirete il cicalino vorrà dire che avete perso!

Compatibile CON TUTTE LE RP

Maker è un termine che, piano piano, è diventato comune nei mass media. Questa parola racchiude i concetti di "smanettone" e "hacker" e ha raggiunto la fama in concomitanza con la diffusione di microcontrollori come Arduino e computer su singola scheda come Raspberry Pi. Usando questi sistemi oggigiorno è possibile realizzare progetti incredibili senza spendere una fortuna e senza dover essere degli ingegneri: dal semplice circuito per accendere a comando un LED a un robot comandato dallo smartphone, tutto è a portata di mano di chiunque di noi. In questo tutorial vi mostriamo, per esempio, come usando pochi componenti è possibile costruire un gioco che metterà alla prova la fermezza della vostra mano. Per la precisione, questo gioco vi richiederà di muovere un oggetto metallico (una graffetta) lungo un filo, senza mai toccarlo lungo tutto il percorso. Toccandolo si perde una delle tre vite che si hanno a disposizione, terminate le quali il gioco si conclude. Ecco cosa vi serve per seguire questa guida: qualunque modello di Raspberry Pi, la versione più recente di Raspbian, una vaschetta di plastica, della creta (o comunque una sostanza modellabile, tipo il pongo), una breadboard, tre LED, quattro resistori da 220 Ohm (caratterizzati dai colori rosso, rosso, marrone), un buzzer (o cicalino), 16 cavetti (11 maschio-femmina, 1 femmina-femmina e 4 maschio-maschio), un pulsante, una cassa audio o un monitor con casse se decidete di usare l'HDMI. E, ovviamente, una graffetta metallica per carta. A RP spenta, collegate tutti i componenti seguendo il layout che vedete nella pagina a fianco. Potete scaricare il diagramma in alta risoluzione dall'URL **http://bit.ly/LXP158_buzzer**. Questo progetto usa una versione aggiornata di Raspbian perché include il nuovo ambiente desktop studiato dai ragazzi della Raspberry Pi Foundation. Questa distro include anche uno dei punti chiave del nostro gioco, la libreria pygame, presente di default sia per Python 2 sia per la versione 3.

Editor e librerie

Iniziate aprendo un terminale e digitando semplicemente il comando seguente:

```
$ sudo idle3 &
```

Dovete usare **sudo** per poter accedere al GPIO della RP, mentre l'**&** alla fine della riga fa partire in background l'editor IDLE3 e vi restituisce il controllo del terminale. IDLE3 dovrebbe apparire sullo schermo dopo qualche istante: dovreste vedere



› Lo scopo di questo articolo è costruire un gioco che metta alla prova il vostro sangue freddo: dovete far viaggiare un oggetto metallico lungo un filo senza toccare il filo stesso. Se non ci riuscite, la Raspberry Pi fa partire una musichetta e mostra un'immagine d'avvertimento

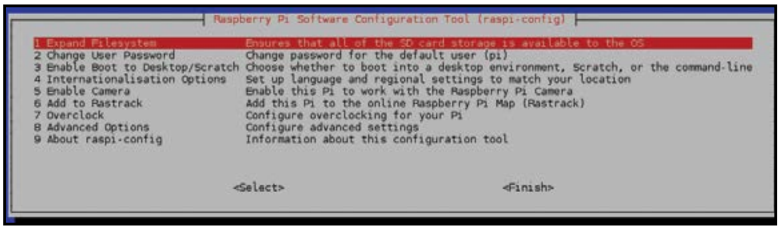
la sua finestra di shell, nella quale potete scrivere i comandi Python 3 e vedere istantaneamente il risultato. La stessa shell è il mezzo tramite il quale osservare l'output di progetti più ampi, ma nel vostro caso avete bisogno di una finestra per inserire del codice, per cui andate in **File** -> **New** per aprire un nuovo documento. Ricordatevi che quando avrete finito di digitarlo, sarà necessario salvarlo prima di mandarlo in esecuzione. Per questa operazione è sufficiente andare nel menu **File** -> **Save** e denominare il file **wire.py**. Vi consigliamo di salvare il codice di tanto in tanto, anche prima di averlo terminato, onde evitare perdite accidentali di dati. Le prime righe dello script servono per importare le librerie necessarie:

```
from time import sleep
import RPi.GPIO as GPIO
import pygame
```

Come prima cosa si importa una singola funzione, **sleep**, dalla libreria **time**; questa funzione viene usata per mettere in pausa il programma in certi momenti chiave. Vi abbiamo fatto importare solo una funzione e non l'intera libreria perché in questo modo non si spreca memoria, visto che le altre funzioni di time non sono necessarie. La seconda riga, invece, importa tutta la libreria Python RPi.GPIO (chiamandola semplicemente

Raspi-config

Se vi siete avvicinati da poco a Linux e alla Raspberry Pi, modificare le impostazioni del piccolo computer potrebbe essere complicato. Fortunatamente, grazie al lavoro dei progettisti della scheda, esiste un tool che vi dà una mano. **Raspi-config** racchiude un insieme di strumenti per la modifica di vari aspetti del funzionamento della RP, come l'overclocking, l'abilitazione dell'accesso via SSH e l'indirizzamento dell'audio al jack da 3,5'' o alla porta HDMI. Per usare questo tool aprite un terminale e digitate **$ sudo raspi-config**. Il suo menu è diviso in nove gruppi, con l'ultimo (**Advanced Options**) che racchiude le opzioni più complesse e particolari. È proprio qui che dovete spostarvi per controllare l'audio, quindi spostatevi con i tasti cursore sulla voce **Advanced Options** e premete Invio. Qui troverete la voce **Audio Options**, entrateci. Nel menu che appare dovete scegliere dove indirizzare l'audio: se avete collegato un monitor alla porta HDMI della RP potete lasciare l'impostazione di default. Altrimenti, se usate uno speaker collegato al jack da 3,5'' dovete attivare la voce **Force 3.5mm jack** selezionandola e premendo Invio. Terminate le operazioni tornate al menu principale e uscite da raspi-config.



GPIO): essa serve a comunicare con i pin del GPIO (*General Purpose Input Output*) della RP. La terza riga, infine, carica la libreria **pygame** che vi servirà per riprodurre la musica e mostrare le immagini sullo schermo. Per poter usare pygame dovete inizializzarla e fare la stessa cosa anche per la funzione mixer; per farlo vi servono le due istruzioni seguenti:

```
pygame.init()
pygame.mixer.init()
```

Ora dovete concentrarvi sul funzionamento del GPIO. È necessario indicare alla RP quali pin del GPIO si vuole usare. Questi pin sono configurabili in due diversi modi:

» **GPIO.BOARD** - questo metodo numera i pin dividendoli in due colonne: una per i pin pari e l'altra per quelli dispari. Il Pin 1 è in alto a sinistra, vicino alla scheda SD, mentre il Pin 2 è subito alla sua destra.

» **GPIO.BCM** - il layout del GPIO scelto dalla RPi Foundation. Divide in modo organico i pin per un uso più efficiente.

Per questo progetto (come abbiamo già fatto in passato) userete il secondo formato, quindi dovete digitare:

```
GPIO.setmode(GPIO.BCM)
```

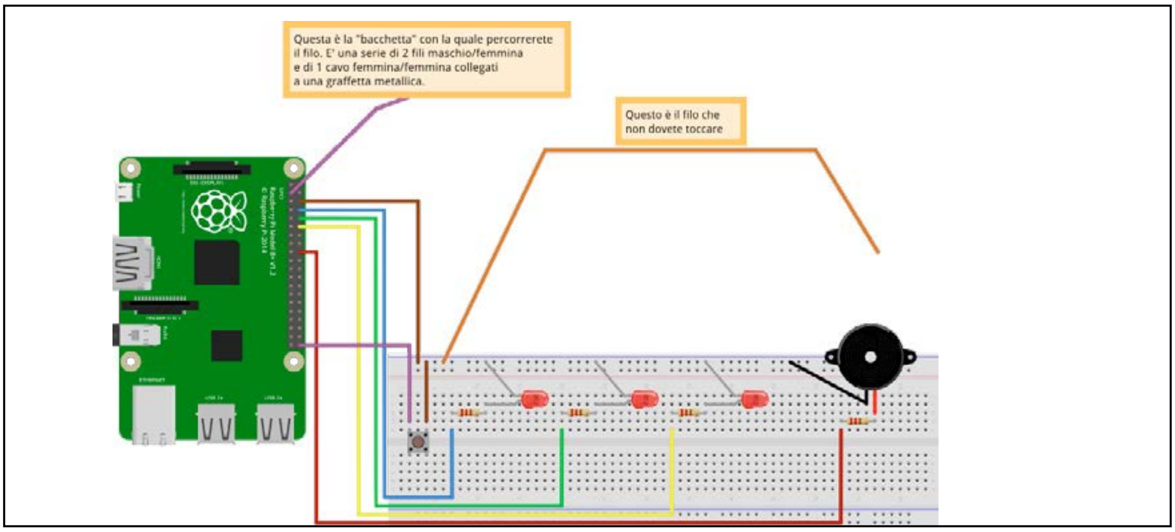
Impostato il GPIO, passate a configurare i pin che verranno usati. Vi serviranno due input e quattro output. Il primo input è collegato al pin 2 del GPIO ed è intenzionalmente impostato al valore alto così che sia attivo. La stessa impostazione viene usate per il pin 21:

```
GPIO.setup(2, GPIO.IN, GPIO.PUD_UP)
GPIO.setup(21, GPIO.IN, GPIO.PUD_UP))
```

Ora passate ai pin di output: userete i pin 14, 15, 18 e 24. I primi tre sono collegati ai tre LED così da creare il contatore di vite del gioco (che, vi ricordiamo, sono proprio tre), mentre il pin 24 è collegato a un piccolo buzzer per segnalare quando la bacchetta viene toccata:

```
GPIO.setup(14, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)
GPIO.setup(24, GPIO.OUT)
```

Bene, a questo punto avete configurato tutti i pin necessari per questo progetto e dovete passare alla scrittura delle funzioni che sviluppano la logica dello script. La prima funzione controlla quanti LED devono essere illuminati, come detto prima, quindi, è il nostro conta vite del gioco. Ogni volta che il giocatore toccherà la bacchetta con la graffetta metallica, gli verrà tolta una vita e la funzione collegata spegnerà uno dei LED. »



› **Per costruire questo schema non serve il saldatore, il che vi consentirà di riutilizzare i componenti impiegati. Userete dei connettori e una breadboard per realizzare un metodo semi-permanente per collegare i componenti**

Tip

Potete scaricare il codice e gli altri file usati dal progetto, compreso lo schema del circuito, puntando il vostro browser all'URL **http://bit.ly/LXP158_buzzer**.

Funzioni di gioco

Iniziate assegnando un nome alla funzione; servirà anche indicare un argomento in input. In questo caso l'argomento è il numero di vite rimaste:

```
def life_counter(i):
```

Scritta questa riga premete Invio e inserite le righe che vi indichiamo tra poco. Ricordatevi di fare attenzione a indentare correttamente le istruzioni usando quattro spazi. Python indica in questo modo le righe che appartengono a una funzione. Userete il costrutto **if ... elseif** che confronta il numero di vite passato come argomento con i numeri 0, 1, 2 e 3: se questo argomento è uguale a uno di questi valori, si eseguono le istruzioni corrispondenti che accendono il numero corretto di LED usando la funzione **GPIO.output** per cambiare lo stato dei pin da off (0) a on (1). La prima condizione da verificare è se il numero di vite è pari a tre (il massimo): se è così si accendono tutti i LED.

```
if i == 3:
    GPIO.output(14,1)
    GPIO.output(15,1)
    GPIO.output(18,1)
```

Se questa prima condizione è falsa, si controlla se il numero di vite è pari a due usando l'istruzione **else if**, che in Python viene scritta come **elif**. In questo esempio possiamo vedere che due dei LED vengono accesi, il 14 e il 15, mentre il 18 viene spento, denotando che vi sono rimaste solo due vite.

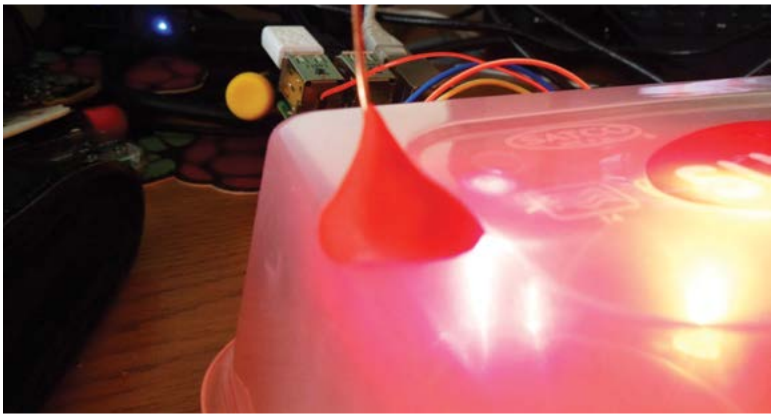
```
elif i == 2:
    GPIO.output(14,1)
    GPIO.output(15,1)
    GPIO.output(18,0)
```

Le due condizioni successive sono simili, stavolta controllando se il numero di vite è pari a 1 o a 0, spegnendo e accendendo sempre il numero corretto di LED. Questo segna la fine della funzione e potete passare alla seconda, che si occupa di mostrare un'immagine a schermo usando la libreria pygame:

```
def picture(img,w,h):
```

La funzione si chiama **picture** e, come vedete, accetta tre parametri in input. Il primo, **img**, viene usato per passare il nome del file immagine da usare, mentre **w** e **h** indicano le dimensioni dell'immagine stessa. Vi mostriamo il corpo della funzione picture, che inizia con la creazione di alcune variabili:

```
pic = pygame.image.load(img)
    background = (255, 64, 64)
    screen = pygame.display.set_mode((w,h))
    screen.fill((background))
```



› **È molto importante attaccare in modo stabile il filo da percorrere perché anche il più piccolo tocco faccia partire l'allarme. Noi abbiamo usato il sugru (è una specie di pongo) per fissare questo filo e isolarlo al contempo dal resto del progetto**

Si usa pic per evitare di ripetere ogni volta la lunga istruzione **pygame.image.load(img)**, usata per caricare l'immagine da usare. Poi si crea un'altra variabile **screen** per impostare le dimensioni della finestra che conterrà l'immagine. Il codice appena scritto prevede anche una tupla (una lista immutabile, cioè una lista che, una volta creata, non può più essere modificata). Essa, chiamata **background**, contiene i valori numerici che specificano il colore di fondo della finestra.
A questo punto è necessario indicare a pygame di mostrare l'immagine caricata:

```
screen.blit(pic,(0,0))
    pygame.display.flip()
    sleep(2)
    pygame.display.quit()
```

Questo è un metodo veloce per visualizzare a schermo delle immagini e viene usato normalmente nei giochi in 2D. Tramite la funzione **flip** si aggiorna l'intera finestra, poi si attende due secondi e infine la si chiude.

Prova e commuta

È venuto il momento di pensare alla funzione principale dello script. Questa parte con il costrutto **try ... except**: in questo modo si terranno sotto controllo le eventuali eccezioni che potranno capitare, si userà un interrupt attivato dalla tastiera più avanti, per terminare l'esecuzione del programma alla loro occorrenza. Le prime tre righe del main creano una variabile chiamata **toggle** e ne impostano il valore a False:

```
toggle = False
    picture('./pi.jpg',640,771)
    while True:
```

La variabile toggle verrà usata per far partire una partita quando l'utente vuole giocare. La seconda riga attiva invece la funzione definita poco fa e le passa come argomento l'immagine da visualizzare, assieme alle sue dimensioni. Infine si crea un ciclo infinito che conterrà la logica del gioco. Essa inizia con le tre istruzioni seguenti, che verificano che sia stato premuto il pulsante d'inizio:

```
        if GPIO.input(21) == False:
            lives = 3
            pygame.mixer.music.load('./fanfare.mp3')
            pygame.mixer.music.play(1)
            0
```

Il ciclo infinito impostato prima serve a tenere sempre sotto controllo lo stato del pulsante d'inizio del gioco. Questo pulsante è collegato al pin 21 (**GPIO.input(21)**), uno dei pin di input che era stato impostato al valore high all'inizio del codice. Quando l'utente preme questo pulsante, il suo stato passa da alto a basso (*Low* o *False*) perché è collegato anche alla massa: premendo si chiude il circuito e il pin 21 assume il valore della massa, 0. Quando ciò succede, il codice procede impostando a tre il numero di vite disponibili e viene suonata la musica d'inizio. Infine si imposta toggle al valore true per indicare l'avvio della partita. A questo punto serve un altro brano di codice che controlla continuamente se lo stato della variabile toggle è "True":

```
        while True:
            if toggle == True:
              print("You have "+str(lives)+" lives left")
              life_counter(lives)
```

Quando la condizione si verifica, lo script stampa a schermo il

numero di vite disponibili e accende (richiamando la funzione apposita creata in precedenza) il corrispondente numero di LED. Sempre all'interno di quest'ultimo ciclo infinito dovete inserire anche le istruzioni seguenti, che verificano lo stato del pin 2 che è un input collegato alla bacchetta (cioè alla graffetta che dovete spostare):

```
if GPIO.input(2) == False:
    GPIO.output(24,1)
    sleep(0.2)
    GPIO.output(24,0)
    pygame.mixer.music.load('./wrong.mp3')
    pygame.mixer.music.play(1)
```

Se lo stato del pin 2 diventa False, viene attivato il buzzer collegato al pin 24 per 0,2 secondi, dopodiché viene riprodotto un file MP3: tutto questo indica che il giocatore ha toccato il filo con la graffetta.

L'ultima condizione

L'ultimo brano di codice Python inizia con le istruzioni per far apparire un'altra immagine sullo schermo; si usa ancora la funzione **picture()** impostata prima:

```
picture('shockdanger.jpg',724,634)
lives = lives - 1
life_counter(lives)
print("You have "+str(lives)+" lives left")
sleep(3)
```

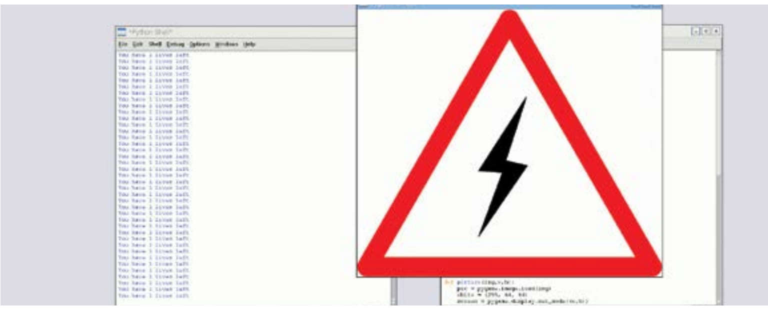
Questo brano, sottrae una vita da quelle disponibili (**lives = lives -1**), mostra tramite i LED il numero di vite rimaste e poi stampa lo stesso numero a schermo tramite l'istruzione **print** per tre secondi, prima di lasciare tornare al gioco l'utente. Ora ci spostiamo alla porzione **elif** del ciclo condizionale:

```
elif lives == 0:
    pygame.mixer.music.load('./wrong.mp3')
    pygame.mixer.music.play(2)
    print("GAME OVER")
    sleep(3)
    break
```

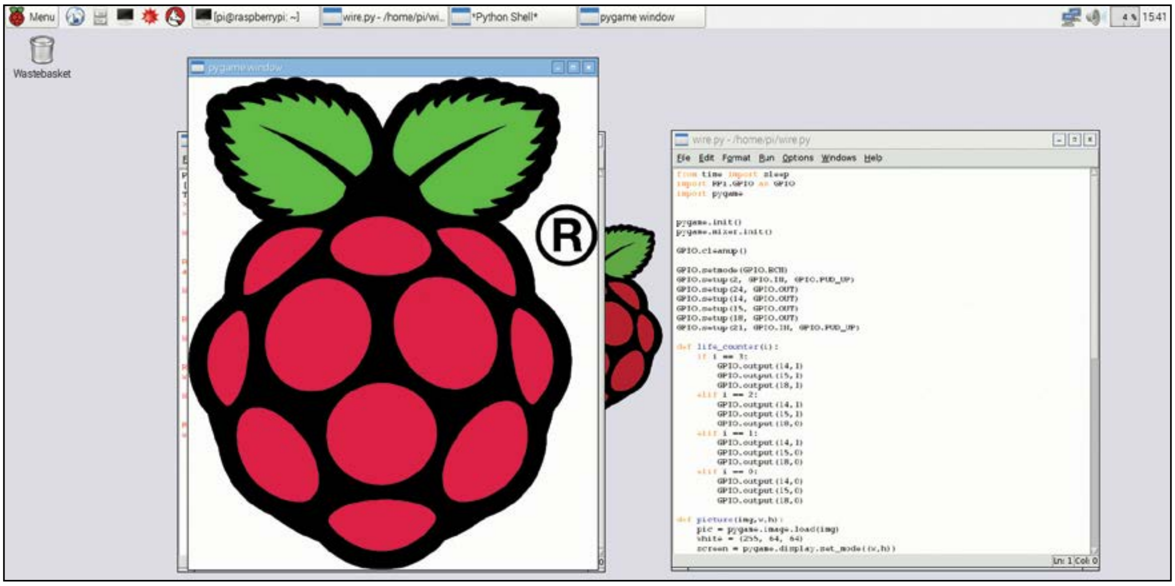
Questa istruzione elif gestisce le operazioni da eseguire quando il numero di vite scende a zero. Per prima cosa viene riprodotto per tre volte il file **wrong.mp3**. Forse vi starete chiedendo perché tre volte se nell'istruzione **pygame.mixer.music.play(2)** c'è scritto 2... Bene, dovete sapere che Python inizia a contare partendo da zero, quindi il totale dei play è pari a tre. Oltre alla musica, il codice stampa a schermo la scritta **GAME OVER** e subito dopo lo script si mette in pausa per tre secondi. A questo punto il programma torna al ciclo infinito principale, attendendo che l'utente prema il pulsante per iniziare nuovamente una partita. L'ultima parte di codice chiude il ciclo condizionale che avevate aperto per controllare lo stato del pin 21, cioè il pulsante per l'inizio della partita:

```
  else:
      print("Press Button to play")
except KeyboardInterrupt:
  GPIO.cleanup()
  print("EXIT")
```

Se il tasto non è stato premuto, il programma chiede all'utente di farlo (**print("Press Button to play")**). La linea successiva va a chiudere il costrutto **try ... except** che avevate visto all'inizio e consente al giocatore di premere la combinazione di tasti **Ctrl**+ **C** per uscire dal gioco. Prima dell'uscita, però, si riporta il GPIO al loro stato normale. Bene, completato il codice ricordatevi di salvarlo per poi mandarlo in esecuzione andando alla voce di menu **Run** -> **Run Module**. Dovreste veder comparire il logo della Raspberry Pi; a questo punto premete il pulsante d'inizio e giocate con cautela: non toccate il filo o perderete una vita. Se non sentite rumori uscite dalla RP e ricontrollate la sua configurazione con raspi-config, come suggerito all'inizio. LXP



› **Un tocco alla bacchetta e si attiva il buzzer sulla breadboard, parte la musichetta e appare il segnale di pericolo sullo schermo**



› **Per mostrare quando il gioco è pronto all'avvio, abbiamo usato il logo della Raspberry Pi come schermata iniziale. Questo è un buon metodo per far capire all'utente quando può iniziare una partita**

L'angolo di Android

News, recensioni e guide sul sistema operativo libero per smartphone

Se hai news da segnalarci o dei commenti scrivici ad angolo_android@linuxpro.it

Novità dall'IFA 2015

Schermi 4K, force touch e batterie maggiorate, aspettando i nuovi Nexus

Si è tenuta come ogni anno l'**IFA** di Berlino, la più importante rassegna europea di tecnologia, il luogo migliore per vedere in anteprima le novità che saranno disponibili per Natale. In realtà quest'anno non sono state presentate innovazioni rivoluzionarie, con i vari produttori che hanno cercato di migliorare i loro dispositivi in un momento in cui per la prima volta le vendite degli smartphone stanno perdendo colpi. Le novità più interessanti sono arrivate da Sony che ha 'triplicato' il suo modello top di gamma **Xperia Z5**, che sarà disponibile in tre versioni: Classica, Compact e Premium. Il modello più interessante dal punto di vista tecnologico è il **Premium**, il primo a presentare un display IPS da 5,5 pollici con la risoluzione di 3.840x2.160 pixel, in pratica un vero schermo 4K. Si può discutere a lungo sull'utilità di avere un display che offre un rapporto di 806 punti per pixel, visto che l'occhio umano non riesce a percepire più 300 punti per pixel, ma rimane il fatto che a livello tecnologico si tratta di un esempio di eccellenza senza paragoni. Una risoluzione così elevata potrà essere sfruttata solo trasferendo le immagini a un televisore o a un monitor 4K oppure eseguendo multipli zoom sul display. L'altissima definizione poi peserà su processore e batteria, mentre le dimensioni di 5,5 pollici sembrano essere il massimo gestibile, come ha dimostrato il mezzo flop del Nexus 6 con il suo schermo da 6 pollici che era difficile anche da mettere in tasca. A nostro avviso è meglio rivolgersi all'Xperia Z5 Classic che integra uno schermo da 5,2 pollici con risoluzione Full HD da 1.920x1.080 pixel anche perché in tutti e tre i modelli della serie Z5 l'innovazione tecnologica arriva dalla fotocamera che può vantare un sensore Exinor da 23 megapixel e un obiettivo ultra luminoso da F=2.0. Non solo, ma l'autofocus è velocissimo (0,03 secondi) e lo zoom digitale a 5 ingrandimenti usa una nuova tecnologia che dovrebbe ridurre al massimo la perdita di qualità. L'altra novità comune ai tre nuovi Xperia è la presenza di un sensore di impronte digitali sistemato sul pulsante di accensione che permette di sbloccare in sicurezza e con la massima velocità lo schermo. Come da tradizione Sony, gli Xperia sono completamente impermeabili al punto da poter funzionare senza problemi anche sott'acqua. Tutto bene quindi tranne che per i prezzi, che partiranno dai 799 euro del modello Premium per arrivare ai 699 del Classic e ai 599 del Compact che, come dice il nome, ha uno schermo da 4,6 pollici e 1.280x720 pixel ma che per il resto ha tutte le caratteristiche dei fratelli più grandi. LXP



› Forma e aspetto squadrati dei nuovi Xperia Z5 sono quelli classici, cambiano invece le componenti hardware che sono quanto di meglio esista oggi in commercio

Lo smartphone pieghevole

Sono ormai anni che si ipotizza la creazione di uno smartphone pieghevole che possa essere arrotolato. La tecnologia esisterebbe pure, ma da qui alla realizzazione pratica ci toccherà ancora aspettare qualche anno.
Nel frattempo dobbiamo rassegnarci ai modelli concavi come il **Flex** di LG oppure al nuovo smartphone **clamshell** di Samsung che dovrebbe venire presentato all'inizio del 2016 e che riprenderà il vecchio design dei cellulari anni '90 con due mini schermi da 3'' ripiegati uno sull'altro e che una volta aperti ne formeranno uno maxi da 6''. Si tratta di un'idea intelligente per evitare di mettere in tasca un dispositivo da 16 cm come il Nexus 6. LXP

ZTE Blade S6

Con l'**S6** ZTE vuole rivivere le magie del Blade originale: esaminiamolo da vicino per capire se ci riesce...

Nel 2010 ZTE ha lanciato l'originale Blade. Aveva uno schermo WVGA da 3,5 pollici, all'epoca di tutto rispetto, e un processore Qualcomm Snapdragon. Il dispositivo ottenne un successo inaspettato, con specifiche che smentivano il suo prezzo conveniente e con il sostegno di una vivace comunità di modding. Giunti nel 2015 ci troviamo di fronte a un nuovo smartphone che potremmo definire l'erede spirituale di quel modello. Molti Blade sono andati e venuti, ma questo è probabilmente il primo ad avere quello speciale mix d'ingredienti che possono rendere un telefono di fascia media un modello top. Ciò che rende speciale l'S6, e che ha reso così l'originale è il rompere con la formula tradizionale adottata da molti suoi rivali. Processore non al top? No, il telefono integra un 64 bit, octa-core Qualcomm Snapdragon 615. Sistema operativo obsoleto con ampie modifiche? Al contrario, il Blade S6 è il primo dispositivo con processore 615 ad avere Android Lollipop. Poca RAM? Non particolarmente, visto che i 2 GB presenti su questo telefono sono lo standard del momento. Poca memoria? Non tanto, sono gli stessi 16 GB del Sony Xperia Z3 (che costa tre volte tanto) e può essere espansa tramite microSD. L'elenco potrebbe continuare. Aggiungiamo subito il Wi-Fi dual band, un'ampia gamma di sensori, un impressionante schermo da 5 pollici, un sensore che troviamo sulle fotocamera Sony di fascia alta, e si capisce subito come l'S6 abbia, per il suo costo, tantissimo da offrire. Naturalmente non ci sono specifiche di punta in ogni suo componente, ma da qualche parte è stato necessario fare dei sacrifici. Lo vediamo principalmente nel design e, in misura minore, nei materiali. Il dispositivo è un tributo palese all'iPhone 6, persino nel "Designed in California" marchiato sul retro della cassa. Questo non vuol dire che il look non sia adeguato. S'impugna bene, proprio come l'iPhone e, anche se completamente in plastica, è di buona qualità, con una minima flessione e delle accattivanti finiture in color argento sul dorso. Senza la necessità di dover isolare l'antenna, il design sembra meno esigente di quanto non sia sul dispositivo Apple. I 7,7 millimetri di spessore permettono di definirlo sottile e, vista la presenza di una batteria da 2,400 mAh, ciò sorprende. Anteriormente è di colore bianco, con tre pulsanti capacitivi blu retroilluminato nell'area inferiore che svolgono le funzioni back, home e menu (una pressione prolungata sul pulsante menu apre la pagina delle app recenti). L'area è dominata dallo schermo da 5 pollici a 720p con tecnologia touch In-cell. È ottimo e molto sensibile. Con il processore veloce e un sistema operativo reattivo e fluido (Lollipop con il suo runtime ART), garantisce ottime esperienze d'uso. ZTE punta molto anche sulla fotocamera del Blade, mettendo in evidenza come il telefono integri una Sony IMX214, che troviamo anche negli Xiaomi MI4, OnePlus One e in altri telefoni economici con elevate specifiche. Secondo ZTE, forse troppo ottimisti, garantisce una qualità da reflex. Comunque, cattura ottime foto, aiutata da un'app fotocamera che può essere usata in modalità automatica o in una configurazione per esperti che offre un'ampia gamma di comandi manuali. Anche la fotocamera anteriore è ottima, con funzioni automatiche per migliorare i ritratti e per scattare solo quando il soggetto sorride. L'S6 ha anche quelle che ZTE chiama "sonorità Hi-Fi". In realtà, il dispositivo ha un solo diffusore collocato sul retro (con dei piccoli piedini che lo tengono sollevato quando posto su una superficie piana) e, anche se l'audio è di buona qualità, non è particolarmente elevato e nessuno è mai riuscito a riprodurre un suono stereo da un singolo altoparlante. LXP

Scheda tecnica

» **Sistema operativo** Android 5.0
» **Processore** Snapdragon 615 octa-core
» **Dimensioni** 144 x 70,7 x 7,7 mm
» **Memoria** 16+2 GB di RAM
» **Peso** 134 grammi
» **Schermo** 5"
» **Risoluzione schermo** 1.280x720 pixel
» **Espansione** microSD



LINUX PRO Giudizio

ZTE Blade S6

Produttore: ZTE
Web: www.ztedevice.it
Prezzo: 250 €

Caratteristiche	10
Autonomia	8
Prestazioni	9
Qualità/prezzo	8

» *Ottimo prezzo, peccato solo per la bassa risoluzione dello schermo.*

Il voto di Linux Pro **9**



Se leggi Linux Pro, ti possono piacere anche...



Completa la tua collezione ordinando gli arretrati a soli € 5,90 cad.

su www.linuxpro.it/arretrati

oppure utilizzando il modulo qui sotto

SCEGLI L'ARRETRATO CHE VUOI ORDINARE

SE VUOI ORDINARE VIA POSTA O VIA FAX, COMPILA QUESTO COUPON

Ritaglia o fotocopia il coupon, invialo in busta chiusa a: **Sprea S.p.A.** Via Torino, 51 20063 Cernusco s/n (MI), insieme a una copia della ricevuta di versamento o a un assegno. Oppure via fax al numero 02.700537672. Per ordinare in tempo reale i manuali collegati al nostro sito **www.linuxpro.it/arretrati.** Per ulteriori informazioni puoi scrivere a **arretrati@linuxpro.it** oppure telefonare allo 02/87158224 tutti i giorni dalle 14.00 alle 18.00

INSERISCI I CODICI delle pubblicazioni che desideri ricevere:

	€
	€
	€
	€
Totale Ordine	€

SCELGO IL SEGUENTE METODO DI SPEDIZIONE:
Indica con una ✗ la forma di spedizione desiderata

	Per **una rivista** spedizione tramite **posta tradizionale** al costo aggiuntivo di	**€ 3,90**
	Per **due o più riviste** spedizione tramite **Corriere Espresso** al costo aggiuntivo di	**€ 7,00**
	TOTALE COMPLESSIVO	€

Data ____ Firma del titolare ____

NOME ____
COGNOME ____
VIA ____
N° ____ **C.A.P.** ____ **PROV.** ____
CITTÀ ____
TEL. ____
E-MAIL ____

SCELGO IL SEGUENTE METODO DI PAGAMENTO (Indica con una ✗ quello prescelto)

- ☐ Versamento su **CCP 99075871** intestato a **Sprea S.P.A. ABBONAMENTI Via Torino 51 20063 Cernusco Sul Naviglio MI** (Allegare ricevuta nella busta o al fax)
- ☐ Bonifico intestato a **Sprea S.P.A. Abbonamenti** sul conto **IBAN IT05 F076 0101 6000 0009 9075 871**
- ☐ **Carta di Credito** N. ____
  (Per favore riportare il numero della Carta indicandone tutte le cifre)
  Scad. ____ CVV ____

____
Nome e Cognome del Titolare della carta (può essere diverso dall'abbonato)

☐ CartaSi ☐ postepay ☐ MasterCard ☐ VISA ☐ Diners Club International ☐ American Express

Recensioni

Tutte le novità in campo software e hardware testate e valutate ogni mese dai nostri laboratori

Se vuoi segnalarci qualche novità scrivi a recensioni@linuxpro.it



Una breve legenda

Ogni test di questa sezione è accompagnato da un giudizio che riassume con quattro indici numerici le principali qualità dell'applicazione o del prodotto hardware messo alla prova. I laboratori di Linux Pro assegnano un voto da 1 a 10 alle seguenti categorie:

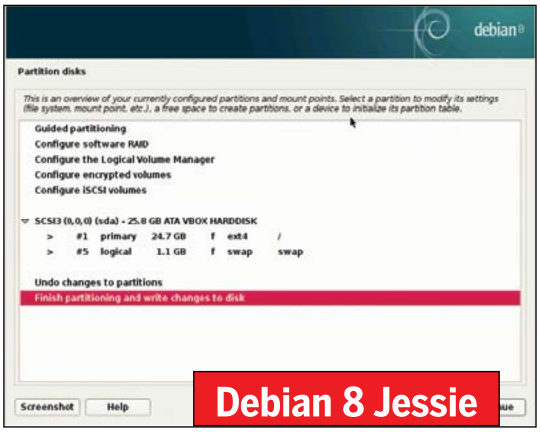
Caratteristiche: fornisce tutte le funzioni di cui abbiamo bisogno? È innovativo?
Prestazioni: esegue in maniera efficiente le sue funzioni? È veloce e affidabile?
Facilità d'uso: dispone di un'interfaccia grafica chiara e facilmente fruibile? La documentazione che lo accompagna è sufficientemente completa ed esaustiva?
Qualità/prezzo: ha un prezzo competitivo? Vale i soldi richiesti per il suo acquisto?

Il nostro giudizio viene poi riassunto da un voto finale, espresso anche graficamente. Ecco la legenda dei voti:

10 Nulla da eccepire. Un prodotto praticamente perfetto.
8-9 Un buon prodotto. I pochi difetti presenti non sono gravi.
6-7 Compie il suo lavoro ma necessita di ulteriori sviluppi.
5-4 Deve migliorare prima di raggiungere un voto sufficiente.
1-3 Un completo disastro. Gli sviluppatori devono tornare alla fase di progettazione.

Ricordiamo infine che i software citati nelle sezioni Confronto e Da non perdere sono spesso presenti nel DVD sotto la voce "Rivista" sotto forma di codice sorgente o binario.

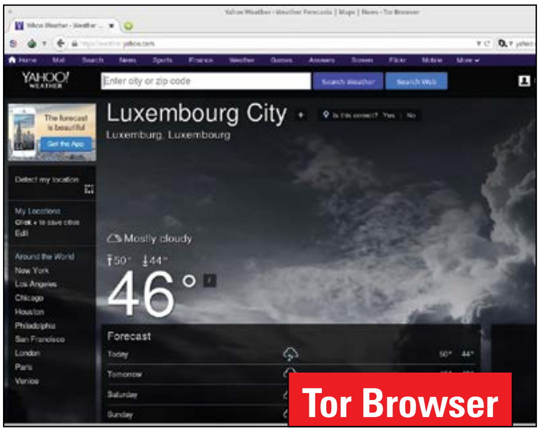
QUESTO MESE...



Debian 8 Jessie



PC-BSD



Tor Browser

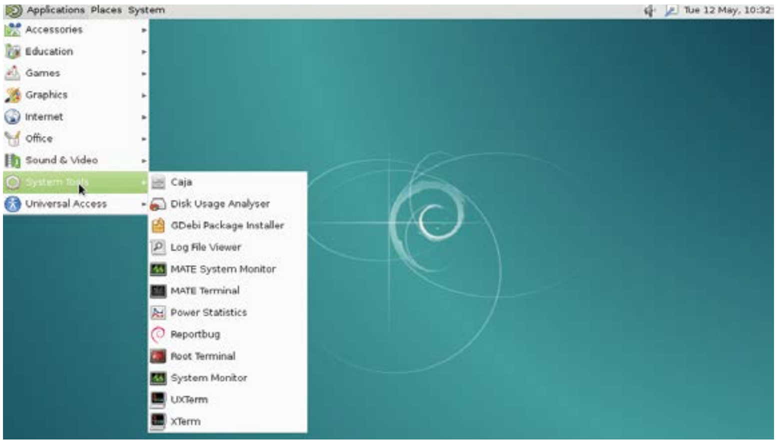
Debian 8 Jessie

Avviamo l'ultima versione della veneranda distribuzione Debian per un classico giro di prova con tanto di sosta ai box per il cambio gomme

In breve

» L'ultima edizione di una delle più prestigiose distribuzioni di Linux. Dopo due anni di lavorazione, eccola con aggiunta di vitamina D... o meglio, Systemd.

Sono veramente passati due anni dall'ultima release stabile di Debian, all'epoca dei kernel 3.2, di SysVinit e Gnome 3.4? In effetti è così; e da allora molte cose sono cambiate. Quando Debian Wheezy è passato dalla versione Testing a quella stabile (trasformandosi così in Debian 7.0), è iniziata la lavorazione di una nuova versione Testing (**Jessie**). Jessie si basava a sua volta sulla distribuzione Unstable, nota con il nome di Sid. Mentre ogni release Testing riceve un nuovo nome in codice ispirato a Toy Story (nome che viene quindi trasferito alla versione Stable alla release successiva), la versione Unstable viene sempre denominata Sid. La nuova versione Testing si chiama Stretch e quella successiva si chiamerà Buster. Tanto per divertirci, l'abbiamo installata su un computer che sarebbe stato all'ultimo grido nel 2006: un Athlon 64 x2 da 2,4 GHz con 2 GB di RAM e scheda video Radeon X1650 AGP. La parte iniziale dell'installazione è filata via liscia, tra impostazione del fuso orario, utenti, eccetera. Lo strumento per la partizione manuale del disco potrà intimidire i novellini ma funziona alla perfezione; e comunque c'è sempre un'opzione per la partizione guidata. Accettando le impostazioni predefinite otterrete una grossa partizione ext4 e una piccola



› **Il desktop Mate rievoca ricordi degli anni ruggenti di Gnome 2 (d'altronde, è quello che utilizzavamo sul nostro computer tanto tempo fa)**

partizione di swap, il che andrà benissimo per chiunque non abbia il fegato di definire uno schema di partizione personalizzato. Il programma taskel vi offre alcune alternative relative agli ambienti desktop e all'impostazione dei server SSH, di rete o di stampa e simili. L'ambiente desktop predefinito è Gnome 3.14 ma basta un click per rimpiazzarlo con Xfce, Cinnamon, Mate, KDE 4 o LXDE. Se avrete a disposizione una connessione di rete al momento dell'installazione, non dovrete che scaricare la prima immagine disco. I pacchetti rimanenti potranno essere scaricati una volta approntato il sistema base. Se invece non avete questa possibilità, potrete scaricarvi fino a sette dischi per l'installazione non in linea dei pacchetti. Inizialmente avevamo attribuito la lentezza mortale dell'installazione dei pacchetti base (circa un'ora) all'antichità del nostro dispositivo; a quanto pare, però, l'andatura tipo Ent dell'installazione è stata notata anche da altri utenti (alcuni dei quali utilizzavano presumibilmente dispositivi più moderni). Abbiamo deciso di indagare più a fondo e abbiamo rilevato lo stesso identico problema anche su una macchina virtuale: lo scaricamento dei pacchetti è rapidissimo, la loro installazione è lentissima. Mmm... Non importa, se tutto va bene dovrete sorbirvela una sola volta. Una volta ultimata la procedura, abbiamo rilevato che tutto quanto funzionava, salvo che Cinnamon non gradiva la nostra antiquata scheda video e si era rifugiato nella modalità di renderizzazione software. Un'installazione base con Mate e Cinnamon occupa poco meno di 4 GB e comprende una gamma accettabile (ma tutt'altro che spettacolare) di applicazioni desktop: LibreOffice, Gimp, Inkscape, Rhythmbox, Transmission, Iceweasel e Icedove, tanto per citarne alcune. Le ultime due non sono altro che Firefox e Thunderbird senza il marchio

Caratteristiche



Solido come una roccia

Il collaudo, intensivo come non mai, garantisce la totale assenza di bug ed errori nell'esperienza Debian Stable, qualunque sia l'impiego del dispositivo.



Systemd

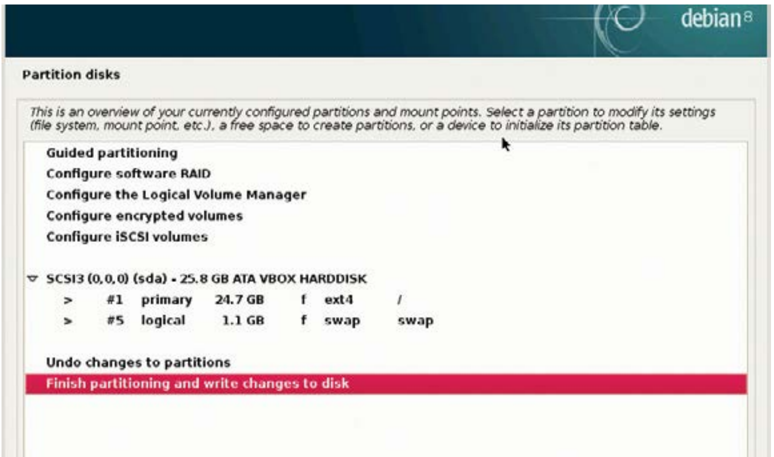
Le urne hanno decretato l'adozione di un sistema init radicalmente nuovo. Potete utilizzare lo strumento preseed per ripristinare il vecchio SysVint oppure decidere di optare per la 'd'.

ufficiale, che non può essere redistribuito. I diversi ambienti desktop, inoltre, sono provvisti di specifiche applicazioni e utility. Sono presenti anche alcune provvidenziali nuove versioni: Python 3.2 (la versione predefinita è ancora Python 2.7), Perl 5.20, PHP 5.6 e Mesa 10.3. Installando il pacchetto build-essential otterrete la versione 4.9 del pacchetto GCC e tutto ciò che occorre per compilare pacchetti sorgente. Diversamente da Fedora, Arch e Mageia, Debian non ha adottato i nomi persistenti per le interfacce di rete, perciò dovrete lavorare con eth0 e wlan0; d'altronde c'è chi preferisce comunque fare così. Il supporto per SSL 3.0 è stato disabilitato per impostazione predefinita nell'intero ecosistema Jessie.

Ecco a voi Systemd

Non abbiamo ancora citato il colosso binario del sistema init. Con il passaggio a Systemd, Debian modifica per la prima volta il suo sistema init. Il nuovo sistema è stato implementato con la caratteristica cura e dovrebbe garantire una totale compatibilità retroattiva con gli script SysVinit (la vecchia struttura ***/etc/rd*.d/directory*** rimane invariata), perciò in teoria l'utente medio non dovrebbe nemmeno notare che qualcosa è cambiato. Tuttavia, basta lanciare $ pstree per scontrarsi con la dura verità: è Systemd a controllare tutto quanto. Si può avere la tentazione di fare della filosofia, domandandosi che cosa sarebbe accaduto se Debian avesse scelto di passare a Upstart (quasi certamente, per esempio, Ubuntu non avrebbe mai adottato Systemd); ma è inutile piangere sul latte versato. Non abbiamo rilevato alcun problema nel setup iniziale di Systemd in una nuova installazione; ma alcuni utenti che

› L'interfaccia di installazione può non apparire granché ma serve allo scopo... anche se nelle nostre prove è risultata estremamente lenta



hanno aggiornato il loro sistema danno notizia di varie incompatibilità. Per la maggior parte si tratta di problemi relativamente secondari a cui è possibile ovviare effettuando la migrazione degli eventuali script init SysV verso i file unit di Systemd. Può essere una sfacchinata ma la sintassi molto più ordinata vi faciliterà notevolmente l'esistenza. Esitiamo però a consigliare Debian ai novellini: anche installando uno degli ambienti desktop principali, infatti, non vi troverete le chicche offerte per esempio da Linux Mint o da elementary OS. La wiki di Debian offre un'ottima documentazione, che tuttavia utilizza in parte un linguaggio che rischia forse di intimidire i profani. Analogamente, Debian avrà difficoltà a conquistare gli utenti con il pallino della modernità: la serie kernel 3.16 utilizzata da questa release è uscita un anno fa, l'ambiente desktop predefinito è Gnome 3.14, la base Extended Support Release (ESR) Firefox per Iceweasel è la 31.6... e l'elenco potrebbe continuare. E se queste versioni non vi sembrano attempate già ora, tenete conto che per la maggior parte rimarranno le stesse per altri due anni. D'altra parte, Debian è celebre anche per la sua configurabilità, perciò non sarete costretti a tenervi le vecchie versioni. Sono disponibili backport dalla versione Testing, che potete attivare aggiungendo un paio di righe a ***/etc/apt/sources.list*** così da poter utilizzare pacchetti più recenti. Se però siete alla costante ricerca di nuove versioni dei software, vi converrà migrare direttamente alla versione Testing (cioè Stretch) o perfino alla versione Unstable, se vi sentite coraggiosi. L'"aggiornamento" (in senso puramente numerico) effettuato in questo modo va benissimo; ma ricordate che indietro non si torna. Non è possibile infatti effettuare il "downgrading" dalla versione Testing alla versione Stable, né dalla versione Unstable alla versione Testing. Anche se preferite una distribuzione Linux più moderna, sarà una buona idea conservare un'installazione di Debian su una partizione distinta, in attesa del lancio di Shiny Linux. Se la caverà tranquillamente per almeno tre anni; e chissà, potreste perfino scoprire l'amministratore di sistema che c'è in voi e decidere di metterci le mani sopra. LXP

Debian: una storia lunga 22 anni

Nei suoi 22 anni di vita, Debian ha dimostrato la sua robusta costituzione e rimane tuttora la distribuzione preferita da molti utenti avanzati. Questa resistenza è in parte dovuta al ciclo biennale e alla ben coordinata supervisione della Foundation; ma è anche il frutto della decisione di blindare il kernel e le versioni di altri pacchetti principali prima del lancio. Il team Linux Kernel mantiene vari rami a lungo termine che risalgono fino alla versione 2.6.32, perciò il backporting delle soluzioni relative ai bug e alla sicurezza rientra nella procedura standard. Anche i pacchetti core vengono trattati nello stesso modo, il che, insieme all'enorme volume di lavoro svolto dal team addetto ai pacchetti di Debian, permette di mantenere stabili, sicuri e sincronizzati l'uno con l'altro circa 40.000 pacchetti ufficiali. Oltre che per la sua longevità e per la sua stabilità, Debian è famoso per il suo Contratto Sociale, che ne definisce i principi di centralità dell'utente, condivisione con la comunità, sviluppo aperto e impegno a favore del Software Libero. Il software soggetto a licenze i cui termini non sono considerati liberi da questo contratto, come il plug-in Flash, viene relegato nel repository non libero. Per abilitarlo dovrete modificare la vostra configurazione apt. Dovrete procedere in questo modo, per esempio, se deciderete di installare driver video proprietari.

LINUX PRO Giudizio

Debian 8 Jessie

Sviluppatore: The Debian Foundation
Web: https://debian.org
Licenze: Varie

Caratteristiche	9/10
Prestazioni	8/10
Facilità d'uso	7/10
Documentazione	9/10

» *I pacchetti saranno anche vecchiotti ma la distro ne prevede già il supporto fino al 2018.*

Il voto di Linux Pro **8**

Dell Chromebook 11

È il Chromebook più robusto che abbiamo mai visto... ma questo basta a renderlo interessante?

Specifiche

- **CPU:** Intel Celeron Bay Trail-M N2840 dual-core a 2,6 GHz
- **GPU:** Intel HD Graphics
- **RAM:** 4 GB di RAM (DDR 3L, 1600 MHz)
- **Capacità:** unità a stato solido da 16 GB
- **Schermo:** HD da 11,6 pollici, 1366 x 768 pixel, touchscreen
- **Fotocamera:** Webcam a 720 p
- **Connessioni:** Bluetooth 4.0, 802.11ac, Wi-Fi dual-band
- **Porte:** HDMI 1.4, 1 porta USB 3.0, 1 porta USB 2.0, lettore schede, lettore schede secondario con blocco, jack cuffia/ microfono da 3,5 mm
- **Dimensioni:** 297 x 218 x 21 mm (1,32 kg)

Specificamente costruita per sopportare i maltrattamenti tipici dell'ambiente scolastico, la versione di quest'anno del **Chromebook 11** di Dell si caratterizza per la sua robustezza, racchiusa in una livrea piuttosto anonima. Con il suo stile sobrio, il Chromebook 11 assomiglia più che altro a un portatile da ufficio. A causa della sua struttura blindata, il Chromebook 11 risulta leggermente più grosso e pesante rispetto ai prodotti concorrenti. Il peso del Chromebook 11 è più o meno quello di un dispositivo da 11 pollici; rimane comunque abbastanza leggero da poter essere portato in giro per tutto il giorno da uno studente. A distinguere il Dell dagli altri prodotti sul mercato è la sua struttura ultra-robusta. Secondo Dell, il nuovo modello ha superato i test standard dell'esercito USA relativamente a durata, resistenza alla sporcizia, alla polvere, alla pressione, alla temperatura, all'umidità, agli urti e alle vibrazioni, "così da poter sopportare la vita dello studente". Per garantire questi livelli di robustezza, Dell ha inserito il suo Chromebook 11 in un solido involucro in plastica nera, una soluzione di progettazione che serve ottimamente a nascondere graffi, sporcizia e impronte delle dita. La fascia di gomma intorno al bordo del Chromebook 11 contribuisce a proteggere il notebook in caso di cadute. Su questo Chromebook lo schermo touchscreen costituisce un'espansione opzionale, che offre allo studente un altro modo per interagire con i contenuti del suo Chromebook, oltre che servendosi della tastiera e del trackpad. Un'altra caratteristica che distingue questo nuovo Chromebook 11 dai modelli rivali è il perno rinforzato del coperchio. Questo non solo consente di aprirlo con un'angolatura massima di 180 gradi, che facilita la collaborazione tra più utenti; secondo Dell è anche estremamente robusto. Dato che spesso gli studenti tendono ad afferrare e a trasportare il Chromebook tenendolo per il coperchio, la struttura del perno riduce le sollecitazioni e ne previene la rottura. Il perno ospita inoltre potenti altoparlanti rivolti verso l'utente.





› Non capita spesso di imbattersi in un Chromebook 'militarizzato' capace di sopravvivere all'ambiente scolastico...

Compagni di scuola

La tastiera, non illuminata, è sorprendentemente piacevole da usare. I suoi tasti sono quasi a grandezza naturale (circa 1,5 mm più stretti rispetto a quelli di un MacBook Pro da 15 pollici) rendono agevole la digitazione. Inoltre sono estremamente reattivi, anche se la dinamica non è quella della tastiera di un portatile vero e proprio. La tastiera è impermeabile, così come il trackpad. Se il suo processore dual-core Celeron Bay Trail-M non offre le prestazioni dell'Intel Broadwell utilizzato dal Google Chromebook Pixel di quest'anno (molto più costoso), il Chromebook 11 se la cava comunque ottimamente. In condizioni d'uso reali, le prestazioni del Chromebook sono paragonabili a quelle di un Toshiba Chromebook 2. I due modelli condividono infatti processore, RAM e risoluzione a schermo. Mantenendo aperte una quindicina di schede in tre diverse finestre del browser, non abbiamo rilevato alcun rallentamento. In generale è un dispositivo rapido e scattante, sul quale non abbiamo riscontrato ritardi di sorta. Il display del Dell non è IPS come quelli che troviamo sempre più spesso in altri modelli di Chromebook, come il Toshiba, perciò l'angolazione visiva utile è limitata. Inoltre, la sua luminosità media fatica a fare fronte all'illuminazione in esterni. L'audio, d'altra parte, è molto potente e nitido. In condizioni d'uso reali, con una luminosità dello schermo intorno al 60%, la batteria ha resistito per poco più di otto ore di normale utilizzo domestico (navigazione Web, e-mail, elaborazione di documenti e qualche video su YouTube). Una novità interessante è costituita dal fatto che Dell offre una soluzione pratica per la gestione in movimento denominata Dell KACE K1000, che consente di gestire gli inventari, nonché un carrello porta-computer in grado di ospitare in sicurezza e ricaricare fino a 30 dispositivi. LXP

LINUX PRO Giudizio

Dell Chromebook 11 (2015)

Sviluppatore: Dell
Web: www.dell.com/uk
Prezzo: 270 €

Caratteristiche	9/10
Prestazioni	9/10
Facilità d'uso	9/10
Documentazione	9/10

» Dell ha riunito funzioni da portatile da ufficio, dando vita a un prodotto per il mercato scolastico eccellente.

Il voto di Linux Pro: 9

Fedora 22

Abbiamo provato l'ultima release della popolare distribuzione Gnome e abbiamo scoperto che non è più riservata ai soli utenti avanzati

In breve

» L'ultima edizione della distribuzione sponsorizzata da Red Hat ma guidata dalla comunità. È disponibile in tre versioni diverse: l'edizione desktop standard si chiama Workstation. Questa distribuzione continua a offrire novità ideali per attirare gli utenti meno esperti senza scontentare quelli più avanzati. Vedi anche: OpenSUSE e Korora.

Per la sua ultima release, Fedora aveva abbandonato il suo tradizionale ciclo di collaudo semestrale, introducendo numerose novità sul piano organizzativo e sistemico. La più importante di esse è stata rappresentata dalla pubblicazione di tre edizioni distinte: Workstation, Cloud e Server. L'altra novità di rilievo è stata la presentazione di Fedora.next, denominazione ufficiale di Fedora Project per il prossimo decennio. La release più recente della distribuzione segna il ritorno al ciclo semestrale, mantenendo però le tre edizioni distinte. Abbiamo deciso di recensire l'edizione Workstation, progettata per gli utenti domestici ma perfettamente adatta anche ai professionisti.

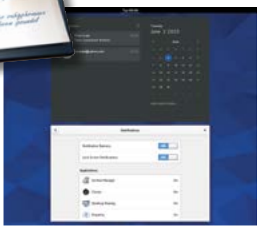


› *Fedora 22* **poggia sulle solide fondamenta gettate dalla release precedente e presenta alcune novità e utili miglioramenti**

All'avanguardia

Come banco di prova per Red Hat Enterprise Linux (RHEL), Fedora ha tutte le carte in regola per offrire ai nuovi utenti tecnologie all'avanguardia. Da sempre questa distribuzione è equipaggiata con i software più recenti e la nuova release non fa eccezione. Gnome Display Manager (GDM) è ora impostato in modo predefinito sul server grafico Wayland invece che su X.org. Sebbene la sessione predefinita di Gnome continui a utilizzare X, con ogni probabilità la prossima release di Fedora si servirà di Wayland. La distribuzione comprende Gnome 3.16, che presenta numerosi miglioramenti visivi e diverse nuove funzioni. L'applicazione **Files**, la schermata di login, la visuale **Activities** e vari altri elementi del desktop hanno ora un aspetto rinnovato. **Boxes**, l'applicazione predefinita per la gestione delle macchine virtuali e remote, presenta tra l'altro un assistente alla creazione di box rinnovato e vari miglioramenti nelle finestre di dialogo relative alle preferenze. L'applicazione **Software** ha fatto molta strada dalla sua introduzione in Fedora 20. Ora vi permette di installare anche elementi aggiuntivi, come font e codec, oltre agli aggiornamenti e ai nuovi software. **Fedora 22** è la prima release che non utilizza Yum come gestore dei pacchetti. Si serve invece di DNF (un fork di Yum), che offre funzioni analoghe e utilizza gli stessi repository di sempre per i pacchetti RPM. DNF può contare tuttavia su un resolver migliorato per le dipendenze, hawkey; e dal momento che DNF è compatibile tramite riga di comando con Yum, la maggior parte dei comandi di Yum, quali per esempio install, groupinstall, remove, ecc. funzionano nello stesso modo anche in DNF. Spesso Fedora è stato definito poco adatto ai neofiti ma il sistema di installazione migliorato Anaconda è decisamente accessibile. Il fatto che i pulsanti siano posizionati nella parte superiore dell'interfaccia di installazione continua a risultarci inspiegabile ma il programma di installazione in sé è altrettanto valido di quelli alternativi presenti in altre distribuzioni. Chi utilizza già Fedora apprezzerà i vari miglioramenti visivi e le nuove funzioni introdotte in questa release. Se vi servite già di una distribuzione basata su RPM, troverete Fedora 22 abbastanza robusto, stabile e veloce da invogliarvi a sostituirlo alla vostra distro. LXP

Caratteristiche



Notifiche

Le finestre a comparsa, i cosiddetti Banner, sono stati trasferiti nella parte superiore della schermata.



Gestore dei pacchetti

Dandified Yum, abbreviato in DNF, è un fork di Yum che costituisce il gestore predefinito dei pacchetti.

LINUX PRO Giudizio

Fedora 22 Workstation

Sviluppatore: Fedora Project
Web: www.fedoraproject.org
Licenze: Varie

Caratteristiche	9/10
Prestazioni	9/10
Facilità d'uso	10/10
Documentazione	9/10

» *La robusta dotazione di software e l'accessibilità degli strumenti fanno di Fedora 22 una distro completa.*

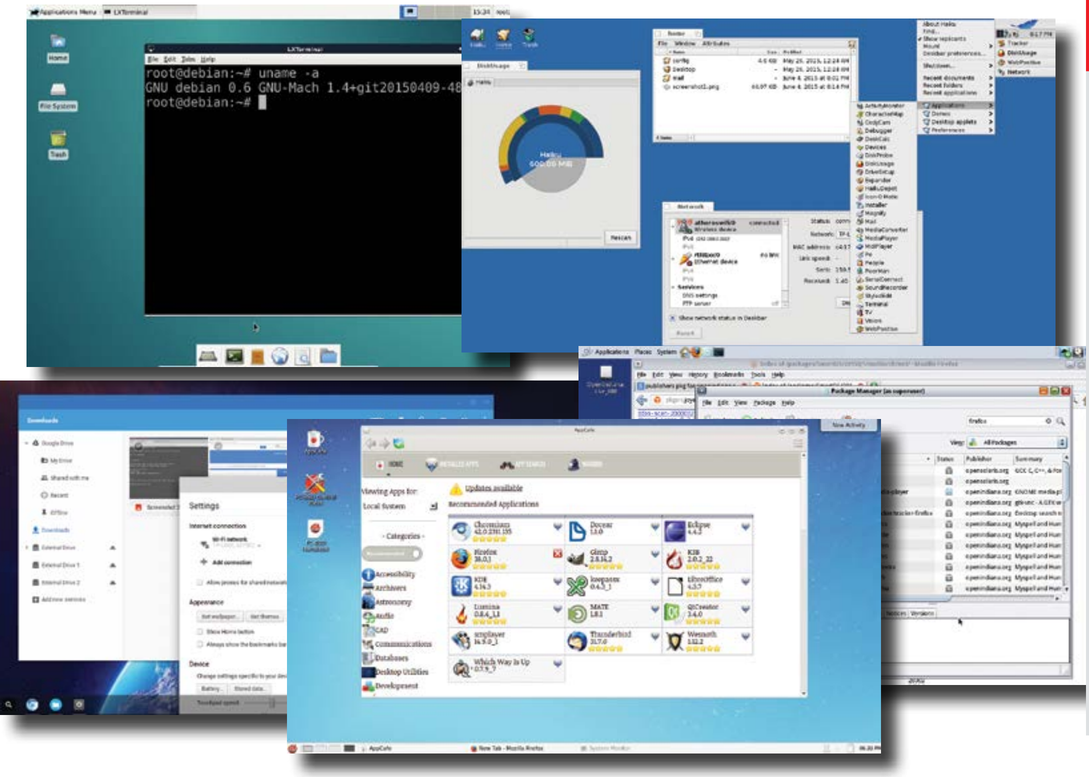
Il voto di Linux Pro 9

Il confronto

» Ogni mese mettiamo a confronto prodotti e programmi per farvi scegliere al meglio!

Sistemi alternativi

Amiamo Linux in tutte le sue varianti, ma è importante ricordare come non sia l'unico sistema Open Source disponibile. Scopriamo cinque alternative



Modalità del test

Il modo con cui si provano determinati sistemi operativi può influenzare significativamente l'esperienza finale. Ecco perché abbiamo deciso di ridurre al minimo questo rischio ed essere quindi il più oggettivi possibile. Per ogni piattaforma abbiamo usato sia un computer reale (un portatile HP con CPU AMD dual-core e scheda grafica Radeon) sia una macchina virtuale (Virtual Box). Abbiamo valutato i cinque concorrenti in base a prestazioni, facilità d'uso, funzioni disponibili, applicazioni, supporto online e stato di sviluppo. La buona riuscita di una prova dipende molto dalla disponibilità dei driver. Per evitare problemi, quindi, vi consigliamo di installare i sistemi prima di tutto in una macchina virtuale. Solo in seguito a un'attenta valutazione della compatibilità, eseguite una migrazione su disco.

La nostra selezione

- » Chromium OS
- » Debian GNU/Hurd
- » Haiku
- » OpenIndiana
- » PC-BSD

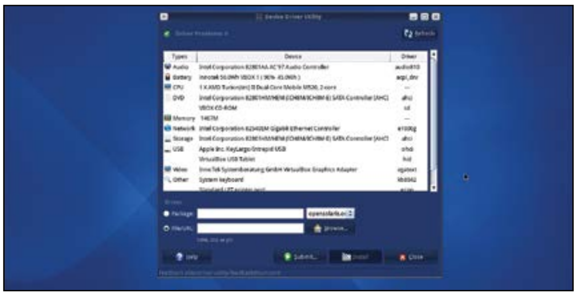
Il panorama di sistemi operativi Open Source che non usano il kernel Linux è molto ampio. Probabilmente vi starete domandando perché valga la pena perdere mezza giornata per provarli. In primo luogo, può essere molto utile per ampliare le proprie conoscenze. In seconda battuta, è importante ricordare come parecchi sistemi siano molto indicati per usi specifici. Per esempio, **OpenIndiana** permette di utilizzare alcune funzioni di storage a livello aziendale. **PC-BSD**, invece, ha tutti i vantaggi di FreeBSD ed è perfetto per le soluzioni Web server. **Haiku** è un progetto unico e non può essere messo a confronto con alcuna distro Linux attualmente in circolazione. Tuttavia è molto veloce e versatile. **Chromium OS** è la versione Open di Google Chrome OS che sta alla base dei sempre più diffusi Chromebook. Andremo così a valutare se i sistemi Cloud di Google possono realmente competere con altre soluzioni più tradizionali. Infine, abbiamo preso in considerazione anche il progetto puramente Open Source **GNU/Hurd**. Il nostro obiettivo nel valutare questi cinque sistemi operativi è fornire una serie di spunti, così da decidere quale sia la miglior piattaforma da usare per compiti specifici in alternativa a Linux.

> "Il panorama dei sistemi operativi che non usano il kernel Linux è molto ampio"

Hardware

Funzionano con qualsiasi computer?

OpenIndiana può essere lanciato sia da chiavetta USB sia da DVD. Molto probabilmente non avrete alcun problema a far funzionare i principali componenti di sistema. Se così non fosse, potete dare un'occhiata alla lista di compatibilità (**http://wiki.openindiana.org/oi/Components**). Qui abbiamo anche trovato un riferimento a un driver proprietario Nvidia utile al funzionamento di alcuni chip. I chipset Radeon, invece, sono supportati tramite i driver VGA di base. Non dovreste avere alcun problema neppure per il riconoscimento della maggior parte dei moduli Wi-Fi. PC-BSD fornisce il miglior supporto hardware e sotto questo profilo è molto simile a Linux. Avete binari ufficiali Nvidia e driver Intel utili per l'accelerazione hardware. Non manca poi il supporto per la maggior parte dei chip Gallium3D Radeon. Se comparate le tabelle di compatibilità di PC-BSD con quelle di FreeBSD vi accorgerete che ci sono davvero poche differenze. Il motivo è molto semplice: usano lo stesso kernel. È quindi possibile installare PC-BSD in un vero PC e aspettarsi che tutto l'hardware funzioni senza alcun problema.
Ci sono comunque alcune eccezioni, tra cui Nvidia Optimus che tuttora non viene supportato. Chromium OS sfrutta un kernel Linux e pertanto riesce a gestire la maggior parte delle periferiche in modo accettabile. Ci sono però alcune limitazioni imposte dal sistema basato su Cloud che peraltro incide sulle funzionalità. Per esempio è del tutto assente il supporto per il touchpad. Considerando altri aspetti, come quello delle periferiche esterne, dobbiamo invece fare un plauso a questa piattaforma. Infatti, è in grado di riconoscere la maggior parte dei dischi fissi, chiavi USB e stampanti collegate. Haiku deve essere considerato sotto un'altra prospettiva. Il sistema può contare su due soli sviluppatori che lavorano a tempo pieno sul codice. Non possiamo quindi pretendere che il supporto hardware sia allo stesso livello dei concorrenti. Tuttavia, Haiku ha dimostrato di sapersi comportare molto bene con i principali moduli Wi-Fi con cui ha avuto a che fare. Le peggiori prestazioni le abbiamo ottenute con GNU/Hurd. Non c'è alcun supporto AGP GART e quindi quasi tutti i chip video utilizzano il driver VESA. Il sistema non usa più di 1,7 GB di RAM e ha uno scarso supporto per le periferiche USB.



› ***OpenIndiana*** **dispone di una pratica utility che permette di mostrare quali driver state attualmente usando**

Verdetto

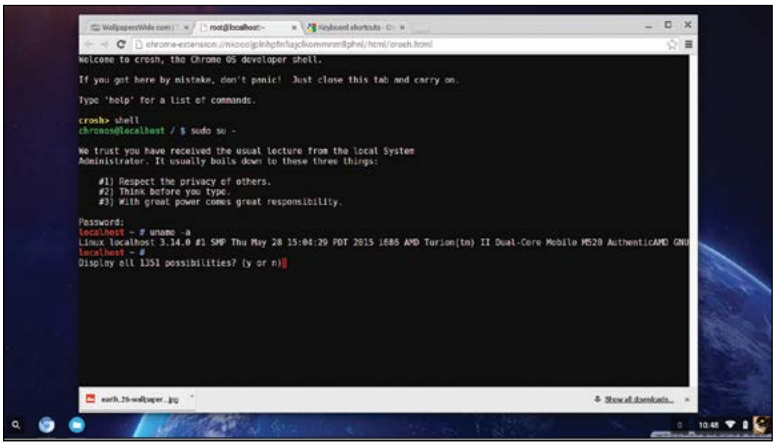
Chromium OS
★★★★★
Debian GNU/Hurd
★☆☆☆☆
Haiku
★★★☆☆
OpenIndiana
★★★★☆
PC-BSD
★★★★★

» *Provateli tutti, ma sarà meglio lasciar perdere Hurd.*

Semplicità d'installazione

Quanto impieghiamo per metterli in moto?

Tutti e cinque i sistemi presi in considerazione in questo confronto sono semplici da installare in ambiente virtuale. È bastato scegliere se avviarli in modalità Live o per l'installazione. In tal occasione, abbiamo provato la stessa procedura anche su un PC reale, creando un DVD e una chiavetta USB. OpenIndiana mette a disposizione un'immagine per le periferiche USB. Tuttavia, all'atto pratico, si è dimostrata particolarmente complessa da scrivere. Utilizzare una tradizionale immagine ISO è quindi stata l'alternativa migliore (**http://bit.ly/1lfPr1m**). PC-BSD, al contrario, è perfetto per essere installato tramite chiavi USB. Permette di usare un installer facile e veloce basato su Qt. Una volta completata la procedura, l'accesso al sistema avviene in pochi secondi e senza alcun problema. Chromium OS si è invece dimostrato deludente anche in questo contesto. Infatti, può essere lanciato solo tramite un'immagine del sistema operativo che deve essere installata in un drive rimovibile. Abbiamo trovato una guida che permette di montare l'ambiente anche in un disco rigido, ma è importante sottolineare come in questo caso non venga utilizzato alcun vero installer. Si tratta per lo più di script piuttosto semplici. Haiku si è dimostrato il sistema più semplice e veloce da installare. Permette di essere lanciato sia in modalità Live sia per l'installazione. GNU/Hurd, infine, non ha brillato per velocità. C'è voluta circa un'ora per renderlo pienamente operativo.



› ***Chromium OS*** **è semplice da usare, ma non da configurare**

Verdetto

Chromium OS
★★★☆☆
Debian GNU/Hurd
★★★☆☆
Haiku
★★★★★
OpenIndiana
★★★★☆
PC-BSD
★★★★★

» *A parte Chromium e GNU/Hurd, gli altri sono ottimi.*

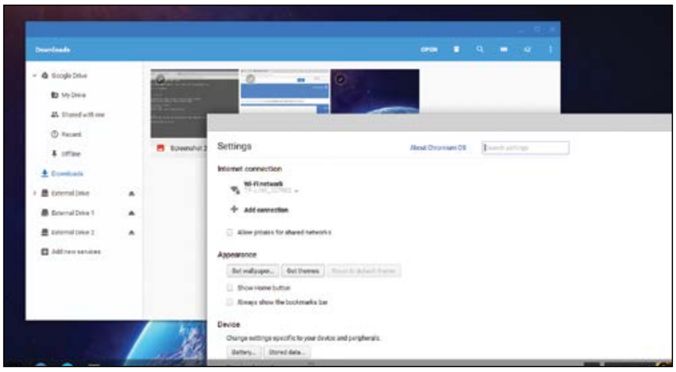
Prestazioni

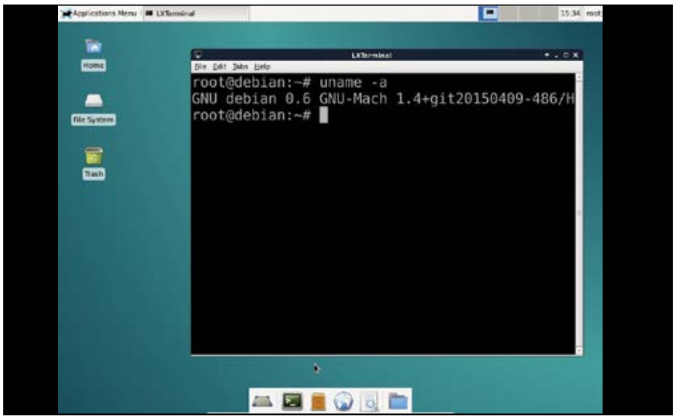
Quanto sono veloci e quanto sono potenti?

Le prestazioni possono influenzare notevolmente l'impressione che abbiamo di un sistema operativo. In questo contesto, quindi, analizzeremo più nello specifico la velocità d'installazione, i tempi di avvio e le latenze che intercorrono tra un click del mouse e il lancio della rispettiva applicazione. Se c'è qualche rallentamento, cercheremo quindi di capire da cosa è determinato. Naturalmente l'esperienza che avrete dipende molto dal tipo di hardware di cui si dispone.

Chromium OS ★★★★☆

Per quanto riguarda Chromium OS non possiamo certo lamentarci. Utilizza il kernel Linux, si basa su Gentoo e sfrutta le versioni più recenti dei vari componenti di sistema, rimuovendo peraltro quasi tutte le applicazioni locali. Dopo l'avvio del server X.org, il resto viene gestito dal browser, inclusi il log-in, la rispettiva sessione, la gestione delle finestre e via dicendo. Chromium OS tenta poi di utilizzare le versioni originali dei pacchetti sorgente Gentoo. Al contempo, sfrutta un significativo numero di patch provenienti dalla comunità di Google, utili a ottimizzare il sistema in alcune piattaforme hardware. Le prestazioni del sistema sono quindi molto buone. La configurazione minima può essere di 1 GB di RAM, con una CPU di fascia bassa.





Debian GNU/Hurd ★☆☆☆☆

Debian GNU/Hurd ha finalmente importato il supporto per X.org. Tuttavia, dal punto di vista dell'utente medio, si è rivelato particolarmente instabile e pieno di bug. In primo luogo, non permette di utilizzare il desktop grafico tramite l'accesso di un normale account. Per parte nostra, siamo riusciti ad avviare LXDE usando $ startx sotto root. Un sistema che usa Hurd, inoltre, si rivela molto lento sia nella reattività sia nell'avvio. In più, non è assolutamente facile capire da dove derivino queste scarse prestazioni. Infatti, non siamo riusciti a comprendere se il tutto fosse causato dalle scarse performance 2D gestite dal driver VESA, oppure da alcuni problemi di I/O. Abbiamo provato a usare **Phoronix Test Suite** per Debian GNU/Hurd, ma i test compatibili con questo sistema si sono rivelati davvero pochi.

Livello e qualità del supporto

Quando c'è bisogno di aiuto a chi ci rivolgiamo?

Così come nel mondo Linux, quando abbiamo bisogno di aiuto è importante sapere a chi rivolgersi. OpenIndiana dispone di un vasto sito Web (**http://wiki.openindiana.org**) che fornisce capitoli dettagliati sull'installazione e l'utilizzo del sistema. C'è però da considerare un aspetto importante: le informazioni contenute si rivolgono soprattutto a sviluppatori e amministratori di sistema. PC-BSD, invece, fornisce un aiuto molto più mirato anche agli utenti e dispone di una splendida comunità di supporto (**www.pcbsd.org/en/community~~number=plural**). Qui è possibile trovare i collegamenti a forum, mailing list, chat IRC, blog e molto altro ancora. Ci sono poi parecchie risorse non ufficiali che fanno riferimento sempre a BSD, nonché i vari contenuti per Free-BSD rilevanti per entrambi i sistemi. Chrome OS dispone di un discreto numero di guide (**www.chromium.org/chromium-os**), incluse le istruzioni per l'avvio rapido. Tuttavia ci sono poche informazioni che vengono condivise con il pubblico generale. Tutto viene ancora tenuto sotto lo stretto controllo di Google. Haiku fornisce un manuale ufficiale, inclusa una guida per sviluppatori (**www.haiku-os.org/guides**). Le informazioni sono abbastanza concise, ma al contempo utili. Infine, la documentazione presente su **www.gnu.org** e **www.debian.org/ports/hurd** è forse il miglior elemento del progetto GNU/Hurd.

Verdetto

Chromium OS
★★★☆☆

Debian GNU/Hurd
★★★★☆

Haiku
★★★★☆

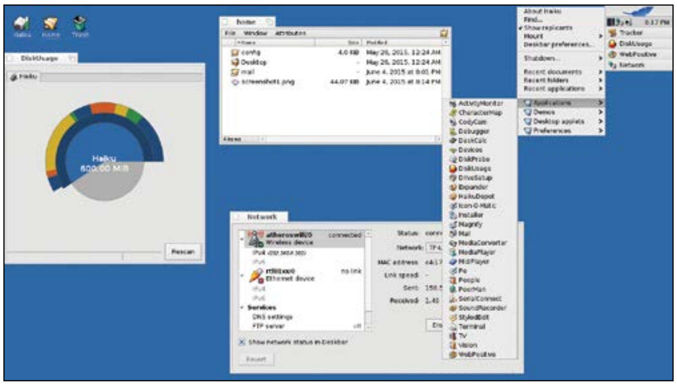
OpenIndiana
★★★☆☆

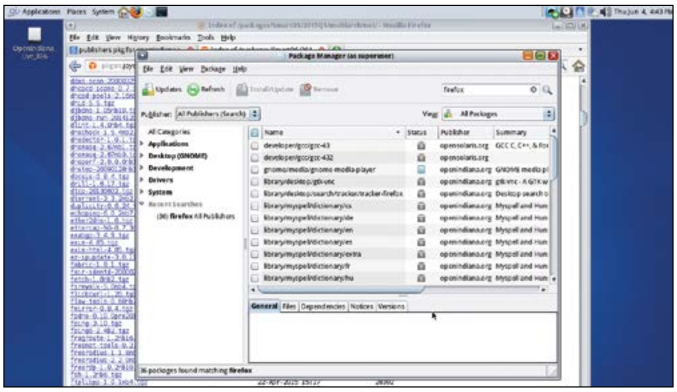
PC-BSD
★★★★★

» *BSD è un ottimo rivale per molte distro Linux.*

Haiku ★★★★★

Nell'altra pagina abbiamo lodato le prestazioni di Chromium OS. Nonostante questo gli abbiamo assegnato solo quattro stelle. Haiku ha invece fatto il pieno. Il motivo è molto semplice: questo sistema è davvero fulmineo e ha una reattività molto più marcata di Chromium OS. Haiku, infatti, ci ha sorpreso su tutti i fronti. Ha mostrato i migliori dati per ciascun task, tra cui 10 secondi per l'avvio e tra 1 o 2 secondi per il lancio di qualsiasi applicazione. Tenendo presente che ci troviamo di fronte a un sistema a 32-bit, con tanto di compilatore obsoleto GCC2 e senza accelerazione grafica, possiamo veramente gridare al miracolo. Haiku, peraltro, può contare su uno stack grafico perfettamente ottimizzato e ridotto ai minimi termini.





OpenIndiana ★★★☆☆

Le prestazioni di OpenIndiana tendono a diminuire nel momento in cui si utilizzano driver video di tipo VESA. Indipendentemente da questo, tuttavia, il sistema ha mostrato una particolare lentezza nella procedura d'avvio, nonché nell'esecuzione delle varie applicazioni. La piattaforma si basa su un filesystem di root ZFS che, pur aggiungendo alcune funzioni molto utili, pesa enormemente sulle prestazioni del desktop. OpenIndiana può sfruttare anche una modalità di funzionamento mista 32/64 bit. Il motivo della sua lentezza è comunque presto scoperto. Il kernel Unix funziona perfettamente solo a 64 bit, mentre la maggior parte dei componenti di sistema ha un'architettura a 32 bit. Per quanto riguarda le prestazioni nella grafica 2D, pur essendo più lente rispetto a Linux, non sono deprecabili.

PC-BSD ★★★☆☆

Una volta installato, per veder comparire la schermata di log-in di PC-BSD ci vuole circa un minuto. L'andamento generale del desktop KDE è piuttosto buono sia con hardware reale sia in modalità virtualizzata (PC-BSD abilita automaticamente le Guest Additions di VirtualBox). A causa del deficit tra ZFS ed ext4, il sistema non è veloce come la maggior parte delle distro Linux. Ciò nonostante si rivela abbastanza flessibile e pratico da utilizzare. Applicazioni come Firefox o LibreOffice vengono lanciate in pochi secondi. Tuttavia, per aggiornare la piattaforma sono necessari svariati minuti di attesa che possono stressare anche la persona più paziente. Per quanto riguarda la maggior parte dei campi d'uso, invece, non abbiamo alcuna lamentela da fare. Possiamo addirittura sfruttare la grafica accelerata.



Stato dello sviluppo

In quanti lavorano allo sviluppo di questi sistemi?

Un sistema operativo gestito in modo attivo è di fondamentale importanza per chi lo utilizza quotidianamente. Ciascuno dei nostri cinque concorrenti ha un numero variabile di persone che lavora al proprio progetto. Questo significa che tra una release e l'altra intercorre sempre un diverso periodo di tempo. OpenIndiana può ancora contare su una vasta schiera di appassionati che hanno lasciato la comunità OpenSolaris. Il ritmo di rilascio attuale, comunque, è molto lento. L'ultima versione è di agosto 2013, esattamente un anno dopo quella precedente. PC-BSD è molto più rapido sotto questo punto di vista. Infatti, il team riesce a rilasciare una nuova release tra i tre e i cinque mesi. Chromium OS, invece, vanta centinaia di programmatori e la sua versione è alla pari con quella del browser. Tuttavia non ci sono immagini ISO ufficiali per il sistema operativo di Google, quanto più un insieme di build create da diversi appassionati. Lo sviluppo di Haiku è estremamente lento. Come detto in precedenza, il progetto conta solo su due sviluppatori in pianta stabile. Ciò nonostante, dimostra una sequenza di rilascio molto costante. GNU/Hurd ha visto invece una nuova release proprio quest'anno.

Verdetto

Chromium OS
★★★★☆

Debian GNU/Hurd
★★★★☆

Haiku
★★★★☆

OpenIndiana
★★☆☆☆

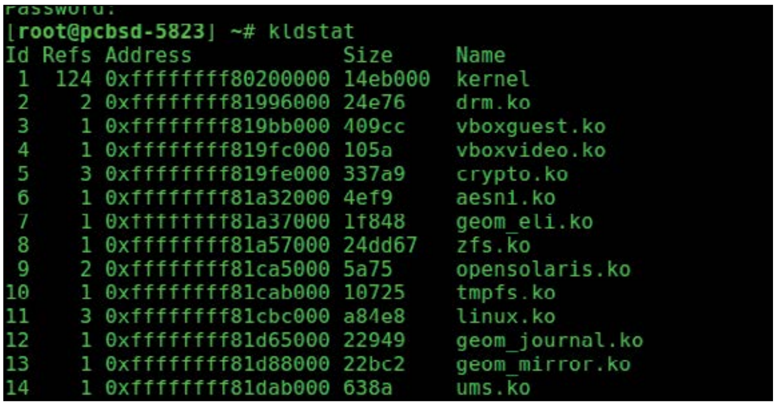
PC-BSD
★★★★★

» Siamo solo un po' preoccupati per il futuro di OpenIndiana.

Funzioni e applicazioni

Quante applicazioni possiamo sfruttare in ogni singolo sistema?

OpenIndiana offre un set di applicazioni desktop già a partire dall'ISO per l'installazione (questa ha un volume inferiore ai 900 MB). Se però volete trovare altri software, potete sfruttare i repo principali e quelli extra disponibili su **http://sfe.opencsw.org** e **http://smartos.pkg.ec**. Tuttavia, è importante sottolineare come in questi ultimi ci sia veramente poco materiale. PC-BSD è dotato di AppCafe, un gateway per il proprio repo di sistema, nonché di una svariata selezione di software che funzionano su FreeBSD (solo a riga di comando). Inoltre, non dimentichiamo la tecnologia kldload che consente a un sistema BSD di eseguire i binari per Linux, tra cui Skype, Flash e altri. Chromium OS può sfruttare solo Chrome Web Store, vale a dire il market per App Web disponibile per Chrome. L'aspetto negativo di questi software è che non possono competere con quelli tradizionali. Infatti, pur non essendo deprecabili, non sono paragonabili alle controparti desktop. Per fare un esempio, non troviamo niente che possa lontanamente essere considerato alla pari di Gimp. Haiku fornisce un insieme piuttosto modesto di software, per lo più accessibili da HaikuDepot, il gestore pacchetti per il sistema. Alcuni siti che mettevano a disposizione programmi per Haiku sono stati chiusi, tuttavia è ancora possibile trovare qualcosa basato su Qt4. Il team di GNU/Hurd, al contrario, è riuscito a importare circa l'80% dei pacchetti Debian. Tuttavia, sul fronte desktop, gli unici disponibili sono Xfce e LXDE.



› **Il comando *kldstat* mostra quali moduli Linux vengono usati da BSD**

Verdetto

Chromium OS ★★☆☆☆
Debian GNU/Hurd ★★☆☆☆
Haiku ★★★☆☆
OpenIndiana ★★★☆☆
PC-BSD ★★★★★

» *In Chromium OS si sente la mancanza delle app desktop.*

Esperienza d'uso

Quanto sono intuitivi? È semplice usarli?

OpenIndiana dispone di una modalità Live che vi accoglie con un desktop Gnome 2.30. Tra le app disponibili, trovate Firefox 10 ma non LibreOffice che invece viene sostituito da OpenOffice. Se quindi non vi sentite a disagio con i vecchi pacchetti di OpenIndiana, non dovreste avere problemi. L'unico vero ostacolo da affrontare sarà far riconoscere la scheda di rete. Se il sistema non riesce a rilevarla automaticamente, allora dovrete procedere con una configurazione manuale (per una guida dettagliata, leggete **http://bit.ly/1SYuPk2**). PC-BSD non dispone di alcuna modalità Live, ma l'unica difficoltà da affrontare per la sua installazione è il partizionamento. I sistemi BSD, infatti, utilizzano una diversa denominazione per le unità. **/dev/sda1** diventa **dev/ada0s1a**. A parte questo, PC-BSD può essere installato senza problemi insieme ad altre distro Linux su partizioni differenti e residenti nella stessa unità. L'installer utilizza KDE4 per impostazione predefinita, ma in AppCafe ce ne sono altri. Approcciare a Chromium OS non è semplice, soprattutto in virtù della sua installazione. Tuttavia, durante l'uso si dimostra particolarmente intuitivo. Se non siete sviluppatori, è possibile ottenere un'immagine pre-costruita da **http://bit.ly/ArnoldtheBat**. È necessario, dopo l'avvio, accedere a un prompt dei comandi con **Ctrl**+**Alt**+**T**. Qui, aggiungete `$ sudo /usr/sbin/chromeos-install` seguito dalla password di root, quindi selezionate un dispositivo di destinazione. Haiku è molto facile da installare e configurare. L'interfaccia ricorda molto da vicino quella di OS X. Se poi riconosce fin da subito la scheda di rete, sarete subito online. Debian GNU/Hurd richiede alcuni passi post-installazione. Per esempio, è d'obbligo eseguire

```
$ dpkg-reconfigure x11-common
```

per permettere agli utenti di avviare una sessione X (date un'occhiata a **http://bit.ly/HurdConfig**).



› ***WebPositive*, il browser predefinito di Haiku, è un'applicazione Web basata su WebKit**

Verdetto

Chromium OS ★★★☆☆
Debian GNU/Hurd ★☆☆☆☆
Haiku ★★★★★
OpenIndiana ★★★★☆
PC-BSD ★★★★☆

» *Haiku può superare quasi ogni distro Linux leggera in circolazione.*

Sistemi alternativi

Il verdetto

L'abbondanza di sistemi operativi Open Source dimostra che una comunità di sviluppatori aperta può fare davvero grandi cose. PC-BSD è il vincitore assoluto di questo confronto. Vanta ottimi risultati in tutte le prove a cui lo abbiamo sottoposto. Il sistema operativo è flessibile, veloce e molto stabile, in più è in grado di riconoscere quasi tutti i componenti hardware in circolazione. Si può usare in modalità Live, ma la versione desktop offre la stessa soddisfazione che ci si aspetta da una distro Linux. Haiku è un sistema operativo che nasce da un contesto modesto. Ciò nonostante dimostra grinta da vendere e probabilmente anche un futuro piuttosto roseo. Ci sono build realizzate con il compilatore GCC2 che può ancora eseguire applicazioni originali BeOS, insieme a software relativamente moderni in Qt4. Lo sviluppo di Haiku non è veloce, ma grazie all'attenzione dei programmatori, il sistema gode di rilasci piuttosto costanti. Non manca poi il supporto per la maggior parte dei moduli Wi-Fi e per le periferiche USB. OpenIndiana sta invecchiando. In passato le speranze riposte in questo progetto erano molte. Tuttavia, allo stato attuale, il lavoro di sviluppo sembra essersi arenato e negli ultimi anni non abbiamo visto alcuna novità rilevante. Chromium OS può essere definito un esperimento che tenta di sostituire i desktop tradizionali con piattaforme basate sulla Cloud. Questo, però, porta con sé diversi svantaggi. Se tutta la vostra attività si basa sull'uso del browser, allora non ci sono problemi. Se però volete andare oltre e ricercare la flessibilità dei programmi desktop, è molto meglio guardare altrove. Nonostante questo, Chromium OS si dimostra veloce ed elegante, nonché piuttosto semplice da utilizzare. Il quinto e ultimo posto, invece, viene riservato a GNU/Hurd che può essere considerato un tentativo di ricreare un sistema GNU puro. Gli sviluppatori Debian hanno fatto un buon lavoro per fornire una distribuzione con microkernel GNU/Mach, ma siamo ancora lontani dall'avere stabilità e flessibilità apprezzabili.



› Con *PC-BSD* non vi renderete conto di usare un sistema diverso da Linux fino a quando non aprite il terminale

> "PC-BSD è veloce, affidabile e in grado di riconoscere quasi tutti i componenti hardware"

1° PC-BSD ★★★★★
Web: www.pcbsd.org **Licenza:** BSD **Versione:** 10.2
» Dopo Linux, la scelta migliore per un sistema Open Source.

2° Haiku ★★★★★
Web: www.haiku-os.org **Licenza:** MIT **Versione:** R1 Alpha 4.1
» Davvero una grande sorpresa. Veloce, flessibile e facile da usare.

3° OpenIndiana ★★★★☆
Web: http://openindiana.org **Licenza:** Per lo più CDDL **Versione:** 151a8
» Un sistema stabile, ma con alcuni limiti.

4° Chromium OS ★★★☆☆
Web: www.chromium.org/chromium-os **Licenza:** BSD **Versione:** 41
» Non è un sistema operativo, ma un browser ricco di funzioni.

5° Debian GNU/Linux ★★☆☆☆
Web: www.debian.org/ports/hurd **Licenza:** GPL **Versione:** 8.1.0
» Nonostante i tanti anni di sviluppo c'è ancora tanto da fare.

A voi la parola...

Non siete d'accordo con le nostre scelte? Avreste usato altri sistemi? Inviate le vostre opinioni su questo confronto a: **recensioni@linuxpro.it**

Considerate anche...

Potremmo includere una vasta gamma di sistemi operativi diversi, tuttavia ci limiteremo a consigliarvene solo tre. Se amate Google, potreste provare **Android X86** (**www.android-x86.org**). Si tratta di un porting non ufficiale di Android che funziona perfettamente su PC desktop. È un sistema operativo basato su kernel Linux e permette di gestire le migliaia di applicazioni Android con la stessa semplicità che riscontrate in uno smartphone o in un tablet. Un'altra opzione da tenere in considerazione è **ReactOS**, un clone Open Source di Windows. Tra i tanti vantaggi che può vantare, possiamo citare la particolare semplicità d'uso e di configurazione. Infine, consigliamo di dare un'occhiata anche a **KolibriOS**, una serie di fork di Darwin che potrebbero riservare tante soddisfazioni. LXP

Da non perdere

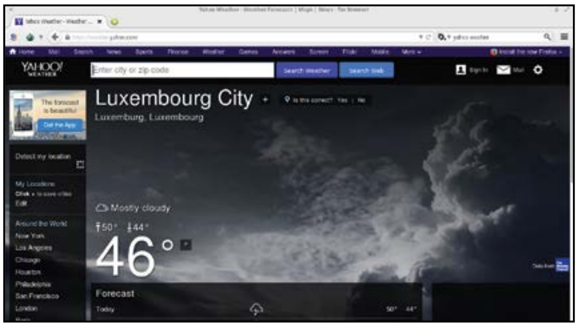
Tor Browser » GNU Nano » ZFSonLinux » Plank » GNU Hurd » QmapShack » Guake » Minilens » MegaGlest » Vera » DNSCrypt

Browser Web

Tor Browser

Versione: 5.0.2 **Web:** www.torproject.org

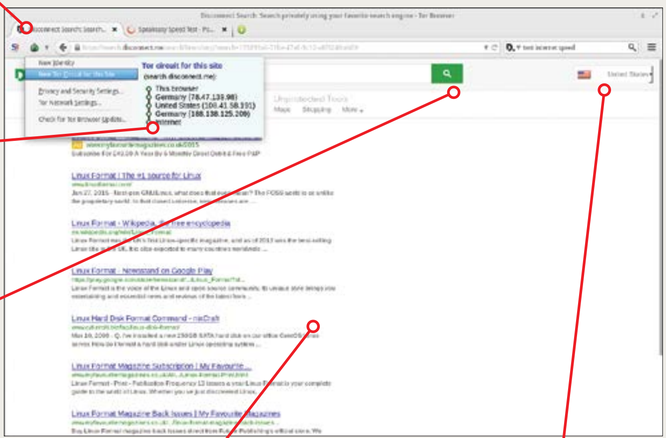
Tor Browser non suonerà certo nuovo a chi è attento alla propria privacy. Noi di Linux Pro siamo soliti citarlo quando consigliamo un programma che permette di garantire un buon anonimato in Rete.
Tor, è importante ricordarlo, non è solo un browser, bensì un sistema che include un gruppo di server capaci di mascherare la posizione di un utente su Internet. Il tutto può essere usato senza avere alcuna esperienza informatica. Infatti, l'intera procedura anonimizzante, avviene in automatico. Il punto di forza di Tor è la capacità di far rimbalzare i dati tra i vari server, senza stabilire un collegamento diretto. In questo modo, non è possibile tracciare l'IP originale della connessione e di conseguenza riuscite a interfacciarvi con Internet senza compromettere la vostra privacy. Da questo punto di vista, Tor può essere usato anche per altri scopi. Infatti, consente di eludere censure basate sulla posizione o sui DNS, consentendo a chi si trova dietro una cortina di ferro di accedere a contenuti altrimenti bloccati. Un esempio calzante può essere l'attuale censura operata dalla Repubblica Popolare Cinese nei confronti di molte pagine e servizi occidentali. Tor Browser è basato su una recente versione stabile di Firefox che peraltro include diversi plug-in ed estensioni predefinite. Tra queste, citiamo **HTTPS Everywhere** che permette di crittografare il traffico di Rete, **NoScript** per disabilitare i JavaScript automaticamente, **Fteproxy** per controbattere eventuali ispezioni del proprio traffico. Tor può collegarsi anche a siti che hanno parametri di accesso estremamente blindati e in un cui il protocollo HTTP è disabilitato. Alla base della sicurezza garantita da questo browser ci sono **obafs2/3/4** e **ScrambleSuit**. La nuova versione dispone poi di alcune interessanti migliorie che contribuiscono a innalzare ancora di più l'affidabilità del sistema anonimizzante. Tor Browser è disponibile per tutti i sistemi operativi e in tutte le lingue, nonché in versione 32 e 64 bit. L'installazione non richiede nessuna competenza. C'è poi una semplice procedura guidata che consente di configurare alcune impostazioni e stabilire una connessione attraverso una catena di nodi.



› **Il meteo di Yahoo non è riuscito a determinare la nostra posizione. Pur essendo in Italia, ci colloca in Lussemburgo**

> **"Tor è uno strumento straordinario per mantenere l'anonimato"**

L'interfaccia di Tor Browser



No scripts
La funzione **NoScript** permette di configurare un'enorme quantità di voci.

Menu Onion
Da qui è possibile accedere alle impostazioni di rete, nonché dare un'occhiata alla catena di server che maschera la posizione.

Ricerca
Il motore di ricerca Disconnect utilizza una VPN. Questo rende possibile mantenere il completo anonimato anche quando si effettuano ricerche. Evitate poi che i siti Web traccino la vostra posizione.

Finestra del browser
Tor ha lo stesso motore di rendering di Firefox. La velocità di connessione, però, è nettamente inferiore.

Scegliete un Paese
I risultati del motore di ricerca variano in base al Paese. Scegliete la vostra posizione geografica.

Editor di testo

GNU Nano

Versione: 2.4.2 **Web:** www.nano-editor.org

Gnu Nano è un semplice ed elegante editor di testo basato su Ncurses che a sua volta si rifà a Pico. Uno dei punti di forza di questo programma consiste nella possibilità di usare una ricca serie di scorciatoie da tastiera per velocizzare il lavoro e la navigazione. La maggior parte è attivabile anteponendo **Ctrl** al tasto da premere. Alcune, però, sono disponibili anche con **Alt**. Per esempio, con **Ctrl**+spazio o **Alt**+spazio si passa di una parola in avanti o indietro, con **Alt**+**G** si balza a una riga identificata con un numero specifico. Potete poi sfruttare una vasta serie di parametri aggiuntivi utili a lanciare Nano con impostazioni personalizzate (nano -h le elencherà tutte). Provate $ nano -t per abilitare l'auto-salvataggio a ogni uscita dal programma, $ nano -B per attivare il backup automatico di tutti i file modificati e $ nano -m per lanciare il supporto per il mouse (utile per la modalità testo con GPM abilitato). La nuova versione garantisce una stabilità che purtroppo mancava da diversi anni. Inoltre, prende le funzioni delle release più vecchie e le migliora sotto tutti i punti di vista. Troviamo un sistema **undo** completamente funzionante (è abilitato per impostazione predefinita), il blocco dei file compatibile con vim, il supporto per la formattazione, nonché l'evidenziatore di sintassi e molte altre cose. La vostra distribuzione Linux potrebbe non essere in grado di fornirvi fin da subito la versione più aggiornata di GNU Nano. Consigliamo quindi di scaricare il tarball dal sito ufficiale del programma. Per installare il software in Ubuntu, assicuratevi di avere il repo **source** abilitato per apt, quindi iniziate con:

```
$ sudo apt-get install build-essential
```

Installate Nano con:

```
$ sudo apt-get build-dep nano
```

Infine compilate il tutto con:

```
$ ./configure && make && sudo make install
```

> **"Un semplice ed elegante editor di testo basato su Ncurses"**



› ***Nano*** **non richiede particolari abilità per essere utilizzato. Inoltre, è oramai l'editor di testo predefinito di molte distro**

Filesystem

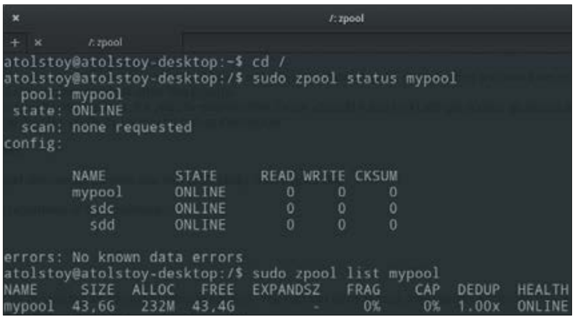
ZFSonLinux

Versione: 0.6.4 **Web:** http://zfsonlinux.org

ZFS è un filesystem combinato con un gestore logico dei volumi (LVM). Progettato da Sun Microsystem, è stato implementato in Solaris. Un filesystem ZFS dispone di parecchie caratteristiche, alcune delle quali uniche. In primo luogo, è in grado di memorizzare fino a 256 zebibytes (Zib) di informazioni organizzate in un massimo di 264 data pool. Uno di questi è invece capace di archiviare fino a 3x 1.023 petabytes di dati. Il supporto per numeri di queste dimensioni deriva dall'architettura a 128 bit di ZFS che viene sempre più spesso utilizzato per attività aziendali. ZFS, comunque, non è stato usato solo in Solaris, ma anche in Linux. Questo filesystem è molto utile nell'*home computing*, in quanto possiede una serie di caratteristiche che lo rendono ottimale per la gestione di questo compito: il controllo di integrità continuo, la riparazione automatica degli errori e molto altro ancora. Utilizzando la funzione snapshot è possibile sfruttare il backup dei file senza utilizzare software di terze parti. Basta solo usare le funzioni integrate nel filesystem. C'è un comodo PPA (**ppa:zfs-native/stable**) che contiene tutti i pacchetti per le principali versioni di Ubuntu. Una volta aggiunto, basta scrivere $ sudo apt-get update, quindi installare **ubuntu-zfs**. Questo permetterà di completare il resto delle dipendenze, nonché compilare il modulo del kernel Linux per il supporto ZFS. Per controllare che tutto funzioni correttamente, richiamate il modulo:

```
$ sudo modprobe zfs
$ lsmod | grep zfs
```

> **"ZFS è ottimo per le operazioni di home computing"**



› **Il gestore logico dei volumi *ZFS* è un'interessante alternativa agli LVM di Linux**

Adesso proviamo alcuni esempi. Create un pool composto da due chiavi USB su **/dev/sdb** e **/dev/sdc**:

```
$ zpool create mypool sdb sdc
```

Per definire ancora meglio l'identificazione, utilizzate l'ID del dispositivo anziché sdx. Lo trovate, usando il comando $ sudo ls /dev/disk/by-id. A questo punto avrete creato e montato un pool nella directory **/mypool**. Per controllarne lo status, usate $ sudo zpool status mypool, quindi per valutare il suo uso: $ sudo zpool list mypool. Adesso potete creare uno snapshot del filesystem con:

```
$ sudo zfs snapshot mypool@snapshot1
```

Dock

Plank

Versione: 0.10 **Web:** https://launchpad.net/plank



› ***Plank* è pratico, veloce e funziona in qualsiasi ambiente desktop attualmente in circolazione**

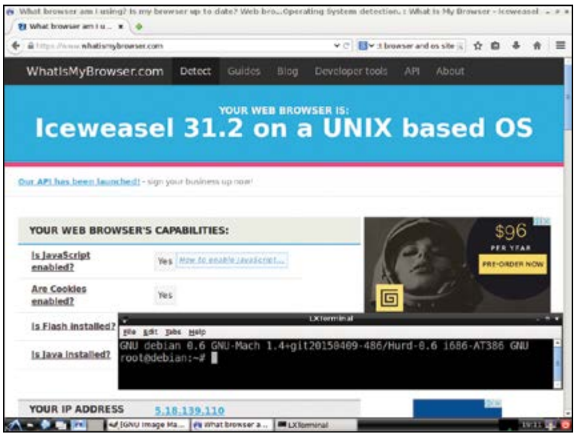
Apple OS X ha da sempre colpito molto per la sua ottima grafica, nonché per il dock estremamente flessibile e ben progettato. Infatti, proprio per questo motivo, molti sviluppatori Linux lo hanno preso a modello per le proprie distro. **Plank** può essere considerato un progetto che punta molto sul fornire un dock semplice e utile per mettere a disposizione solo quello di cui si ha realmente bisogno. Ciò non toglie che le sue funzioni possano essere estese con caratteristiche più specifiche. Plank, inoltre, è la tecnologia su cui si fonda Docky (a partire dalla versione 3.0.0), un'altra applicazione che consente di sfruttare un dock molto pratico. A differenza di Plank, però, Docky mette a disposizione fin da subito una serie di funzioni avanzate. Quasi tutto il codice di Plank è scritto in Vala, inoltre è possibile accedere alle impostazioni del dock direttamente dal menu contestuale che si apre facendo click con il tasto destro del mouse (in alternativa, le opzioni di configurazione sono disponibili anche nel pannello **Impostazioni** di Elementary OS). Nonostante a un primo utilizzo Plank sembri fin troppo semplice, può essere personalizzato con particolare soddisfazione. Infatti, è possibile scegliere la dimensione delle icone, il posizionamento e l'allineamento sul desktop, le opzioni da tenere nascoste e il tema (quella nuovi devono essere archiviati in **~/.config/plank/theme**). L'unica mancanza evidente consiste nell'assenza delle opzioni per le animazioni. La nuova versione ha poi migliorato le prestazioni generali, nonché corretto diversi bug.

> **"Plank consente di usare un dock semplice ma molto personalizzabile"**

Kernel

GNU Hurd

Versione: 0.6 **Web:** www.gnu.org/software/hurd



› **In Internet, un sistema operativo basato su *Hurd* verrà rilevato come una piattaforma Unix**

GNU/Hurd è un progetto di vecchia data che mira a ricreare il kernel Unix. Hurd sta per *Hird of Unix-Replacing Daemons*, mentre Hird è l'acronimo di *Hurd of Interfaces Representing Depth*. GNU Hurd sfrutta un certo numero di server in esecuzione sopra il microkernel GNU Mach. Questi, a loro volta, aggiungono filesystem, protocolli di rete e il controllo di accesso ai file, più altre caratteristiche che normalmente potete trovare in un kernel per sistemi operativi. Può sembrare strano, ma Hurd viene spesso considerato una specie di utopia per la realizzazione di un kernel totalmente libero, capace di unire la filosofia Unix allo spirito del *free computing*. GNU/Hurd 0.6 è il risultato di oltre 18 mesi di sviluppo e mette a disposizione diverse caratteristiche: l'aggiornamento alla release 1.5 di GNU mach, il supporto di base ACPI, il meccanismo di inizializzazione in stile SysV e molto altro ancora. Per provare GNU/Hurd, ovviamente, avrete bisogno di una distribuzione appropriata. Il team di Debian ha aggiornato la versione 8.0 con il supporto per Debian/Hurd.
A tal proposito, è importante ricordare che Hurd fornisce solo un modesto insieme di driver per l'hardware attualmente in circolazione. Questo significa che nonostante il suo funzionamento sia fuori dubbio, potreste non avere a disposizione alcuna funzione audio, supporto USB o OpenGL. Per provare Hurd, quindi, è meglio utilizzare una macchina virtuale che fornisce una maggiore compatibilità con il kernel in questione. Com'è facile capire, quindi, Debian/Hurd 8.0 non è certo una distro pronta per essere messa in circolazione. Tuttavia può rappresentare una buona opportunità per provare un kernel particolare. Durante l'installazione, per far funzionare il server Xorg, è necessario selezionare il layout di tastiera inglese (Stati Uniti). Si tratta di una limitazione nota con cui tuttora dobbiamo fare i conti. Infine, il desktop che vi accoglierà è LXDE.

> **"GNU/Hurd è il tentativo di realizzare un kernel totalmente libero"**

Strumento per la mappatura GPS

QMapShack

Versione: 1.3.0 **Web:** http://bit.ly/QMapShack



› *QMapShack* **può non essere molto intuitivo, ma in compenso mostra informazioni non reperibili in Google o Bing Maps**

Tra le applicazioni Open Source più degne di nota nel campo della mappatura GPS, c'è sicuramente **QMapShack**. Si tratta del successore di **QlandkarteGT**, un potente programma GIS che consente di visualizzare i dati GPS su una varietà di mappe, nonché pianificare itinerari, esplorare l'orografia del terreno e molto altro ancora. L'interfaccia grafica, purtroppo, non è delle più intuitive. Questo nonostante il programma sia rivolto essenzialmente a chi ama l'aria aperta e viaggia sempre con un Garmin, un TomTom o qualsiasi altro dispositivo dotato di GPS. L'applicazione è in grado di importare, visualizzare e modificare i dati GPS codificati in diversi formati, tra cui GPX. In più, supporta le mappe in formato GeoTIFF, così come IMG in vettoriale tipica di Garmin.
La prima volta che si esegue QMapShack, troverete un'interfaccia praticamente vuota. Premendo il pulsante **Instant fix**, si accede però a un insieme di sei mappe predefinite in alta qualità e in formato TMS e WMTS. Questo materiale è senza dubbio di prim'ordine e mostra una ricca serie di informazioni extra che non possono essere trovate su Google o Bing Maps. Il modo più semplice per sovrapporre un percorso su una mappa è utilizzare il menu **File** -> **Load GIS**, quindi scegliere un file GPX (gli smartphone Android possono generare GPX con l'app **My Tracks**, **http://bit.ly/1GTBQi9**). Lanciare QMapShack in Linux, vista l'assenza di binari precompilati, è abbastanza impegnativo. Per Ubuntu e i suoi derivati, è necessario montare il tutto dai sorgenti:

```
$ sudo apt-get install build-essential g++ mercurial cmake cmake-curses-gui gdal-bin libgdal-dev libgdal1h python-gdal libproj-dev qttools5-dev-tools qttools5-dev qttools5-private-dev qt5-default libqt5webkit5-dev qtscript5-dev checkinstall
```

Acquisite quindi i sorgenti:

```
$ hg clone https://bitbucket.org/maproom/qmapshack QMapShack
```

e infine montate e installate con:

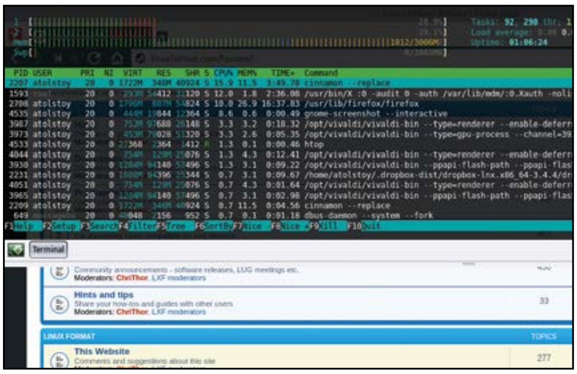
```
$ cd QMapShack && mkdir build && cd build && cmake ../ && make && sudo make install
```

> "Una delle migliori applicazioni Open Source per la mappatura"

Terminale

Guake

Versione: 0.8 **Web:** https://github.com/Guake/guake



› **Per accedere alla riga di comando non è necessario premere alcun tasto. Basta far scorrere il cursore dall'alto in basso**

La quantità di emulatori del terminale di Linux è enorme. Forse viene battuta numericamente solo dagli editor di testo. Per la maggior parte si tratta di console autonome, ma esistono comunque soluzioni molto più particolari. In KDE, per esempio, basta premere **F4** e si apre immediatamente il terminale.
Un altro approccio forse ancora più interessante è proprio quello proposto da **Guake**. Infatti, per aprilo, basta far scorrere il cursore dalla parte superiore dello schermo a quella inferiore. Il programma è un perfetto esempio di integrazione con qualsiasi desktop utilizzato. In più, è estremamente personalizzabile. Guake viene fornito con una ricca serie di funzioni e opzioni accessibili dal menu **Preferences**. Quest'ultimo è separato dal terminale e vi potete accedere senza aprire l'applicazione. Le opzioni sono suddivise in schede dov'è possibile scegliere tutte le funzioni più utili, tra cui i parametri visivi (come la trasparenza) o l'allineamento del testo. La nuova versione dispone poi di un nuovo supporto multi-monitor, ha aggiunto diverse serie di colori e può avviare in automatico uno script prestabilito non appena Guake viene lanciato.
In più, adesso è presente una funzione auto-hide che ha il pregio di supportare i desktop più moderni in circolazione (tra questi anche Unity). Se non avete mai usato Guake prima, potreste trovarvi di fronte al seguente errore:

```
# Guake can not init!
Gconf Error. Have you installed guake.schemas properly?
```

Per risolvere il problema, basta inizializzare di nuovo programma. Richiamatelo premendo **F12**, quindi chiudetelo. I pacchetti di Guake sono disponibili per alcune distribuzioni Linux, tra cui Ubuntu e Arch. Trovate comunque tutto l'occorrente per ottenere l'ultima versione disponibile direttamente nel sito ufficiale.

> "Uno dei più versatili emulatori di terminale che ci siano in giro"

Giochi Applicazioni per il tempo libero

Puzzle platform

Minilens

Versione: 1.0 **Web:** http://alketii.github.io/minilens

Minilens è il nome del robot che controllate in questo titolo. L'ambientazione è post-apocalittica e pone l'attenzione sul prendersi cura del nostro pianeta. Nonostante il messaggio particolarmente importante, il gioco è molto divertente e leggero. Non c'è bisogno di concentrarsi o sudare sette camice per passare da un livello all'altro. Ha una grafica allegra e luminosa che attrae anche i più piccoli. È possibile spostare il proprio robot da destra a sinistra. Tuttavia non si può saltare. Quindi dovrete cercare di muovervi tenendo presente gli ostacoli che si frappongono durante il percorso. Potrete divertirvi affrontando i tanti livelli disponibili che vengono forniti con diversi elementi aggiuntivi, tra cui bombe e scale. Il gameplay di Minilens può essere paragonato a quello di **KGoldRunner** e a molti altri titoli arcade classici che hanno fatto parte della storia videoludica. Per impostazione predefinita, si hanno a disposizione ben otto livelli di tutorial, più 18 extra. Minilens è disponibile sia in versione a 32 bit sia a 64, inoltre offre una procedura d'installazione particolarmente semplice. Il gioco è comunque molto minimalista, non solo per quanto riguarda la parte grafica, ma anche nella fornitura dei file. All'interno dello ZIP, infatti, dovrete solo trovare il file del gioco e quello dei livelli. Minilens sfrutta il motore di **Godot** (**www.godotengine.org**) ed è quindi capace di digerire tutti i livelli personalizzati in formato XML.

> "Il gioco è caratterizzato da una grafica piacevole e molto colorata"



› ***Minilens*** **somiglia molto a** ***KGoldRunner*** **e a tanti altri giochi arcade dello stesso tipo**

Gioco di strategia

MegaGlest

Versione: 3.11.1 **Web:** https://megaglest.org

MegaGlest è un fantastico gioco di strategia in tempo reale. Partito da un progetto abbandonato nel 2009, si è poi evoluto in una fork. Il mondo fantasy è caratterizzato da sette fazioni: Magici, Tecnologici, Indiani, Egiziani, Norvegesi, Persiani e Romani che insieme formano il Megapack. Ogni schiera è dotata di una propria serie di unità, edifici, aggiornamenti, nonché vantaggi e svantaggi che permettono di variare la strategia in base alla fazione che usate. Il tutto, mantenendo un gameplay sempre bilanciato. La fazione tecnologica, per esempio, utilizza guerrieri umani tradizionali, ma presenta dispositivi meccanici di vario genere. Sono molto forti in mischia, ma presentano delle difficoltà a distanza. Le mappe di gioco sono di dimensioni variabili. Si passa da grandi pianure a campi con tanto di fiumi e corsi d'acqua, fino ad arrivare a promontori e montagne. I giocatori devono stabilire i propri insediamenti per ottenere le risorse necessarie allo sviluppo e alla formazione di nuove unità. Come si può capire, quindi, il gioco è fortemente basato sulla saga di **Age Of Empires**. La grafica è particolarmente curata, con tanto di foglie che si muovono e texture ricche di dettagli. La musica non è fastidiosa, ma vi accompagna durante tutto il gioco. Con MegaGlest potete anche creare dei LAN party per divertirvi tra amici. Il titolo funziona nella maggior parte delle distro Linux. Per chi ne avesse bisogno, è comunque disponibile nel repository **Playdeb** (**www.playdeb.net**).

> "MegaGlest è fortemente ispirato alla famosa saga di Age of Empires"



› **Estrarre oro dalle riserve è di vitale importanza per espandersi e creare nuove unità**

Ambiente desktop

Vera

Versione: 1.0.0 **Web:** https://github.com/vera-desktop

Nel mondo Linux non esiste certo mancanza di ambienti desktop. A parte quelli più famosi, ce ne sono molti altri particolarmente flessibili e intuitivi da utilizzare. Uno di questi è proprio **Vera**. Si tratta di una nuova base progettata con GTK3. Ha poi una serie di elementi scritti in Python 3 e Vala, tra cui il centro di controllo, la shell e alcune applicazioni che si auto-avviano al lancio del sistema. C'è però bisogno di alcune integrazioni, soprattutto a causa delle dipendenze di Vera. Per citarne alcune, **PCManFM** (compilato in GTK3), **OpenBox**, **Blueman**, **Screenshot XFCE4** e altri. Vera desktop utilizza poi un proprio pannello (Tint2), un generatore di menu (Alan2) e riesce a fornire una soluzione integrata ben confezionata, con tanto di tema basato su Zukitwo e splendide icone Faba.
Una caratteristica unica di Vera è la procedura di benvenuto a cui ci sottopone. Si tratta di un passo a passo interamente basato su uno script che si distingue per l'eccezionale integrazione con il desktop. Attraverso questa panoramica, vi verranno spiegate le funzioni e i vari scenari di utilizzo, quindi potrete anche cimentarvi in un tutorial che aiuta a prendere fin da subito confidenza con questo ambiente. Parlando di Vera, è però fondamentale considerare anche la sua attuale scarsa diffusione. Naturalmente è possibile installarlo manualmente nella maggior parte delle distro. Tuttavia, se non volete perdervi tra i vari meandri di questa procedura, potete puntare su **Semplice Linux**. Questa distro relativamente moderna e veloce utilizza Vera come desktop predefinito. Semplice è una distribuzione basata su Debian Unstable. Può contare su poche applicazioni preinstallate, ma è una buona soluzione per chi cerca un sistema adatto alla navigazione Web e alla riproduzione musicale.

> **"Vera desktop è scritto con un mix di Python 3 e Vala"**



› **Il Centro di controllo è il componente di *Vera* che può effettivamente vantare più funzionalità**

Proxy DNS

DNSCrypt

Versione: 1.6.0 **Web:** http://bit.ly/DNSCryptProxy

Anche se non è sempre possibile ottenere la massima sicurezza nelle comunicazioni end-to-end tra noi e un sito Web, ci sono diverse soluzioni che possono metterci relativamente al sicuro. Tra queste c'è **DNSCrypt**, un servizio locale che protegge le connessioni tra client e resolver DNS. Una delle ragioni per cui consigliamo di utilizzarlo consiste nella prevenzione dello spionaggio e dello spoofing. Per utilizzare DNSCrypt avrete bisogno di uno strumento chiamato **dnscrypt-proxy**. Questo può essere utilizzato direttamente come resolver locale o come inoltro DNS. A sua volta, autentica le richieste utilizzando il protocollo DNSCrypt, per poi passarle al server. dnscrypt-proxy viene fornito pre-confezionato in Arch, Mageia, Debian Sid e Slackware. Per completare il quadro, vi forniamo una serie di suggerimenti utili a usarlo in Ubuntu e nelle sue derivate.
In primo luogo, prelevate il pacchetto da: **ppa:anton+/dnscrypt**. Per impostazioni predefinita, questo è configurato per utilizzare l'IP **127.0.0.2**, così da non interferire con le impostazioni predefinite della distro (**127.0.0.1**, in Ubuntu, è occupato da **dnsmasq**) . Per avere maggiore sicurezza, i pacchetti utilizzano un account di sistema dedicato e senza privilegi. DNSCrypt eseguirà un chroot nella directory **home** dell'utente e rilascerà i privilegi di root per l'UID di tale account nel più breve tempo possibile. Il resolver predefinito in Ubuntu è **DNSCrypt.eu Resolver #1 @ The Hague, Holland**. Questo, tuttavia, può essere modificato insieme ad altre impostazioni agendo sul file di configurazione **/etc/default/dnscrypt-proxy** (lanciate `$ sudo service dnscrypt-proxy restart` dopo aver modificato il file di cui sopra). Una volta fatto, accedete alle impostazioni per la connessione a Internet in **NetworkManager** e specificate l'indirizzo DNS **127.0.0.2** in **IPv4 Settings**, quindi riavviate la connessione (`$ sudo service networking restart`). Potete verificare l'esecuzione di dnscrypt-proxy con il comando:

```
$ sudo tcpdump -i eth0 dst host 176.56.237.171
```

A questo punto, niente vieta di sostituire **eth0** con il nome della connessione. Ricordate poi che l'indirizzo IP è il resolver DNS usato da dnscrypt-proxy. LXP

> **"DNSCrypt previene lo spionaggio e possibili attacchi di spoofing"**

› **È possibile utilizzare il proxy resolver *DNS* per una maggiore sicurezza**





Tutorial

I nostri esperti offrono i loro consigli di programmazione e di amministrazione del sistema

LA GUIDA DI RIFERIMENTO

Esiste sempre qualcosa di nuovo da imparare in campo informatico, soprattutto in un mondo dinamico come quello di Linux e dell'Open Source. Ogni numero di Linux Pro presenta una sezione dedicata a tutorial realizzati da esperti in moltissimi settori: programmazione, sicurezza, amministrazione di sistema, networking. Troverete informazioni utili sia che siate dei veterani di Linux sia degli utenti alle prime armi. Studieremo con cura anche le applicazioni più diffuse sia in ambito lavorativo che desktop. Il nostro scopo è quello di fornire in ogni numero il giusto mix di argomenti, ma se avete suggerimenti su temi particolari che vorreste vedere trattati, scriveteci via e-mail all'indirizzo tutorial@linuxpro.it

COME RAPPRESENTIAMO LE LINEE DI CODICE

Si presenta spesso la necessità di riportare le linee di codice di un programma. Per favorirne la lettura evidenzieremo le singole linee in questo modo:

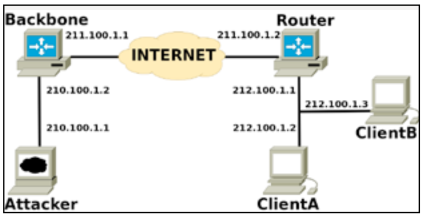
```
begin
  mniWordWrap.Checked := not
end
```

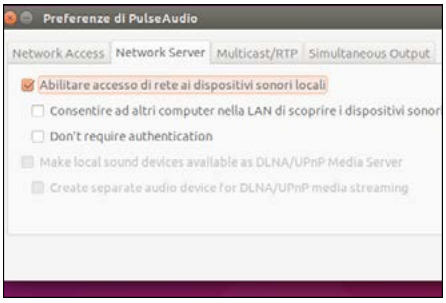
Quando una riga di codice supera la lunghezza della colonna la riporteremo su più righe utilizzando la notazione seguente:

```
printf("Vi preghiamo di inserire
una password.");
```

TUTORIAL







ACCADEMIA DEL CODICE

Sicurezza servizi: rischi e soluzioni

Ecco come testare la propria rete locale per assicurarsi che tutti i servizi in esecuzione siano davvero a prova di bomba

Negli ultimi quindici anni abbiamo assistito, di pari passo con il progresso tecnologico, a un completo stravolgimento nell'uso di Internet e nelle modalità di collegamento alla Rete. Internet e il Web sono divenuti via via una presenza costante e continua nella nostra vita quotidiana: archiviate le connessioni dial up, da attivare al bisogno, il mercato si è orientato verso le cosiddette connessioni flat, attive sempre e pertanto sempre fruibili. Il progressivo, rapido, aumento della banda disponibile in download ha consentito l'estensione della connettività di rete a un numero maggiore di dispositivi, a fronte dell'installazione di un unico punto di accesso alla Rete: una circostanza impensabile qualche anno fa, quando le connessioni erano ottenute per mezzo di modem a 56 Kbps. Con il tempo, ci siamo abituati all'idea di computer connessi permanentemente a Internet, utilizzati per svolgere operazioni sulla Rete (come per esempio il download di file) anche in assenza dell'utente. Si tratta di una consuetudine ormai radicata, che se da un lato comporta notevoli vantaggi in termini di praticità (non ultima la riduzione dei tempi di attesa relativi al download di un file di grosse dimensioni), dall'altro espone le nostre Linux box a minacce aggiuntive, verso le quali è consigliabile adottare le dovute contromisure. In questo numero e nei prossimi, avremo la possibilità di toccare dal vivo i rischi a cui, spesso inconsapevolmente, ci esponiamo nell'utilizzo quotidiano di Internet, e conoscere i possibili rimedi da attuare sui nostri sistemi GNU/Linux per innalzarne il livello di sicurezza complessivo. Scopriremo che, nonostante l'impareggiabile fama del pinguino nel campo della sicurezza informatica, un sistema GNU/Linux mal configurato e/o non aggiornato può risultare vulnerabile al pari di qualsiasi altro sistema operativo proprietario, con l'aggravante di generare nel suo utilizzatore un falso senso di tranquillità.

Virtualbox e l'ambiente di test

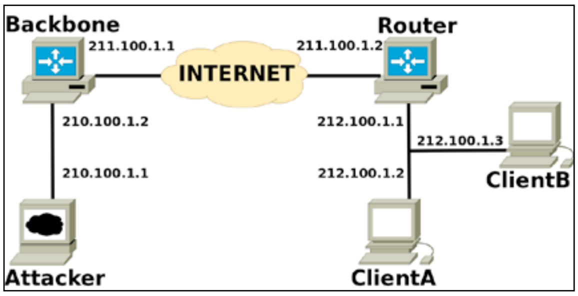
Per raggiungere questi risultati, tuttavia, avremo bisogno di un vero e proprio laboratorio virtuale ove simulare quanto accade negli scenari a rischio. La disponibilità di un tale ambiente di test, se da un lato comporta un onere aggiuntivo sotto il profilo dell'installazione e della configurazione, dall'altro ci consente di sperimentare in libertà, scevri da possibili condizionamenti dovuti a potenziali rischi di sicurezza introdotti sulla nostra macchina fisica. Il primo passo nella realizzazione del nostro laboratorio è rappresentato dall'installazione di un hypervisor, un programma in grado di gestire la creazione e l'esecuzione simultanea di macchine virtuali. GNU/Linux offre molteplici scelte in questo campo, come mostrato ne **Il confronto** pubblicato su LXP 155. In questa serie, in particolare, utilizzeremo **Oracle Virtualbox** che, nei sistemi debian based, può essere installato mediante il comando

```
apt-get install Virtualbox*
```

Mediante il ricorso a Virtualbox, realizzeremo l'infrastruttura di rete schematizzata in (**Fig 1**), ove è rappresentato un tipico esempio di una rete SOHO (acronimo di *Small Office/ Home Office*) connessa a Internet. Nel suddetto schema di rete, infatti, la macchina:

» **Router** funge da router perimetrale della rete LAN;
» **ClientA** corrisponde, così come **ClientB**, a una delle due workstation costituenti la LAN domestica/aziendale, connesse a Internet per il tramite di **Router**;
» **Backbone** rappresenta uno dei router intermedi che consentono l'instradamento dei pacchetti in Internet;
» **Attacker** costituisce la schematizzazione di un generico utente malizioso, dislocato in una qualsiasi area del pianeta e in grado di raggiungere, per il tramite di Internet, i nostri inconsapevoli client.

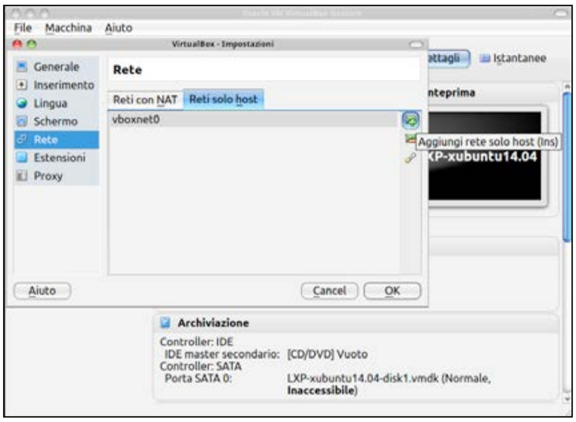
I lettori più attenti avranno notato come, nello schema qui sotto, tutte le macchine siano dotate di indirizzi IP pubblici (vedi box **Indirizzi IP pubblici e indirizzi privati** a pag 65): se tale scelta risulta decisamente plausibile per i router e l'attaccante, può sembrare tuttavia opinabile per quanto attiene i due client. Al giorno d'oggi, infatti, persino i più semplici modem/router (quelli, per intenderci,



› **Fig 1: Lo schema del nostro ambiente di test**

L'autore

Maurizio Russo
Laureato in Informatica presso l'Università "La Sapienza" di Roma, con una tesi sperimentale sullo stack TCP/IP del kernel Linux, è un utente del pinguino dal 2001. Nella sua carriera si è occupato di formazione, sicurezza, networking, progettazione e sviluppo di software.



› **Fig 2: Ecco la schermata di Virtualbox per la creazione di una rete solo host**

consegnatici dal nostro fornitore di connettività a seguito della stipula di un apposito abbonamento per l'accesso a Internet), fanno ricorso al meccanismo del NAT (**Network Address Translation**, vedi box omonimo) per limitare l'utilizzo di indirizzi IP pubblici, provvedendo ad assegnare ai computer serviti degli appositi indirizzi IP privati.
In aggiunta, molti dei più diffusi modem/router integrano al proprio interno delle funzionalità di firewalling, in grado di fornire una prima barriera ai tentativi di intrusione provenienti dall'esterno della LAN. In quest'ottica, lo schema descritto in **Fig 1** potrebbe apparire troppo poco realistico: in realtà, si tratta di un'ottima rappresentazione del cosiddetto *worst case* o "caso peggiore". Per analizzare uno scenario, in informatica si fa spesso riferimento al "caso peggiore", i cui costi di gestione consentono una stima affidabile dello scenario stesso, rappresentando un limite superiore per qualsiasi altra possibile casistica.
Nel nostro caso, esaminare il caso peggiore ci consente di:

» analizzare i possibili rischi a cui possono esporci configurazioni poco accorte (cosa accade, per esempio, se per sbaglio espongo una porta del mio PC utilizzando la funzionalità di port forwarding del modem/router domestico);
» verificare le minacce conseguenti all'utilizzo di gateway dotati di funzionalità di sicurezza minimali (cosa può succedere se il mio PC dispone di un indirizzo IP pubblico, o se utilizzo una vecchia chiavetta per la connessione a Internet);
» comprendere quali siano le contromisure da adottare per incrementare il livello di sicurezza globale dei PC in rete (una contromisura dimostratasi efficace nel "caso peggiore" sarà, a maggior ragione, in grado di operare anche in condizioni di sicurezza migliori).

Installato Virtualbox, e chiariti i motivi alla base dell'architettura di rete proposta in **Fig 1**, possiamo dedicarci alle attività propedeutiche all'installazione delle macchine virtuali. Per garantire la maggiore aderenza possibile della suddetta architettura allo scenario rappresentato (rete SOHO connessa a Internet), è opportuno provvedere a segmentare la rete in tre diverse subnet:

» una per la rete SOHO, comprendente **ClientA**, **ClientB** e l'interfaccia interna di **Router**;
» una per la backbone, comprendente l'interfaccia esterna di **Router** e **Backbone**;
» una per la connessione dell'attaccante (**Attacker**) alla macchina **Backbone**.

Nell'ipotesi di affidare alla macchina fisica il compito di simulare l'attaccante, e tenuto conto della possibili tipologie di connettività offerte da Virtualbox alle proprie macchine virtuali, la segmentazione proposta richiede il ricorso a:

» una rete interna, che denomineremo **intnet**, per la simulazione della rete SOHO;
» una seconda rete interna, **intenet1**, per la simulazione della backbone;
» una rete **solo host** per la connessione dell'attaccante.

Se le reti interne non richiedono alcun onere aggiuntivo, per l'utilizzo delle reti **solo host** è necessario effettuare una breve configurazione iniziale, da eseguire dall'interfaccia principale di Virtualbox. Avviato l'hypervisor, dobbiamo selezionare dal menu **File** la voce **Preferenze**, cliccare sull'icona **Rete**, quindi scegliere la scheda **Reti solo host**. »

Network Address Translation

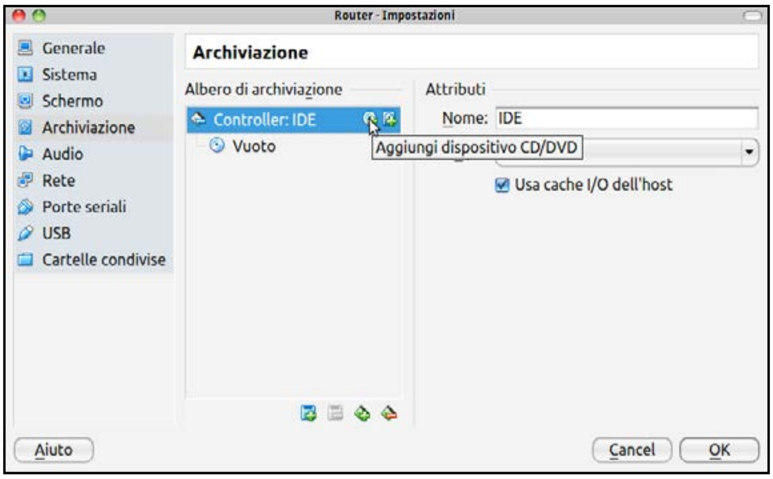
Il *Network Address Translation* (NAT) è una tecnica introdotta dal documento **Request for Comment 1631**, grazie alla quale un numero n di computer, pur dotati di indirizzi IP privati, è in grado di accedere a Internet utilizzando un numero m di indirizzi IP pubblici, con m<n.
Alcune varianti di NAT, come per esempio il *Port Address Translation* (PAT), consentono di ridurre il numero di indirizzi IP pubblici necessari persino a un'unica unità. Si tratta proprio del meccanismo che consente ai gateway casalinghi di connettere a Internet più dispositivi, pur potendo contare su un unico indirizzo IP pubblico (attribuito all'interfaccia del gateway esposta a Internet). Ciò è possibile grazie a manipolazioni operate dal gateway sui pacchetti in ingresso e in uscita alla rete: prima di consentire il passaggio di ciascun pacchetto, infatti, il gateway configurato per effettuare PAT modifica alcuni campi dell'header IP e dell'header TCP.
Per i pacchetti in uscita dalla rete, tali campi sono:

» il campo **Source Address** dell'header IP, in quanto il valore A1 originario (coincidente con un indirizzo privato di uno dei dispositivi della rete) viene sostituito con l'indirizzo IP A2 del gateway (pubblico, e quindi ruotabile su Internet);
» il campo **Source Port** dell'header TCP, in quanto il valore P1 originario viene sostituito con il valore P2 imposto dal gateway sulla base delle porte al momento disponibili.

Ogni modifica apportata a un pacchetto in uscita viene tracciata dal gateway mediante un'apposita tabella, grazie alla quale il gateway è in grado di operare le modifiche inverse sui pacchetti di risposta generati dalle macchine remote. I pacchetti in entrata subiscono infatti la modifica dei campi:

» **Destination Address** dell'header IP, in quanto il valore A2, che identifica l'indirizzo IP pubblico del gateway (l'unico noto al mittente del pacchetto, in virtù della trasformazione operata sui pacchetti in uscita alla rete) viene sostituito, con l'indirizzo A1 originario;
» il campo **Destination Port** dell'header TCP, in quanto il valore P2 imposto dal gateway viene sostituito con il valore originario P1.

Ciò consente, a ciascun dispositivo della rete locale, di stabilire una connessione TCP verso una qualsiasi macchina remota su Internet: da un lato il dispositivo locale invia e riceve pacchetti con il proprio indirizzo IP locale, dall'altro la macchina remota invia e riceve pacchetti con l'indirizzo IP pubblico del gateway, sul quale cade l'onere di operare le necessarie manipolazioni dei pacchetti.



› Fig 3: Cliccando sul pulsante *Controller IDE* è possibile aggiungere un lettore CD virtuale alla VM

Un click sul pulsante **Aggiungi rete solo host** (identificato dall'icona recante una scheda di rete sormontata non a caso dal simbolo **+**) ci consente di creare, sulla macchina fisica, un'interfaccia di rete virtuale denominata **vboxnet0** (**Fig 2**). Facendo click su di essa, è possibile visualizzare una finestra di dialogo composta da due tab:

» nella prima, denominata **Scheda**, dobbiamo inserire 210.100.1.1 come indirizzo IPv4 e 255.255.255.0 come netmask;

» nella seconda, denominata **Server DHCP**, occorre verificare che il flag **Abilita il server** non sia selezionato, in quanto per i nostri esperimenti faremo ricorso esclusivamente a indirizzi statici, come appare chiaro dallo schema in **Fig 1**.

A questo punto la nostra interfaccia di rete **solo host** è pronta. Nel proseguo, **vboxnet0** sarà utilizzata per il collegamento della macchina fisica (che simula l'attaccante) alla macchina virtuale **Backbone**, per il cui tramite, a configurazione completata, la macchina fisica potrà raggiungere i client della nostra rete SOHO.

Le macchine virtuali

Terminate le operazioni preliminari, possiamo finalmente dedicarci alla creazione delle macchine virtuali che simuleranno i computer della nostra rete. Per limitare i requisiti hardware, e ridurre all'osso i tempi di preparazione dell'ambiente di test, faremo ricorso (oltre che alla macchina fisica per la simulazione dell'attaccante) a quattro macchine virtuali basate su distribuzioni live. Tale decisione ci consente di evitare gli oneri connessi con l'installazione dei sistemi operativi e, abbinata a una saggia scelta delle distribuzioni, di contenere sensibilmente la quantità di RAM richiesta alla macchina ospitante.
Non a caso, ricorreremo alle distribuzioni:

» **Damn Small Linux** (la cui ISO può essere scaricata all'URL **http://www.damnsmallinux.org/download.html**) per simulare il router di backbone (macchina **Backbone**) e uno dei due client (**Client1**);

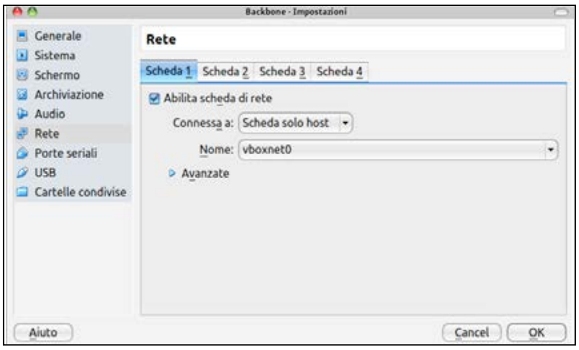
» **Xubuntu 14.04 LTS** (la cui ISO può essere scaricata all'URL **http://xubuntu.org/getxubuntu**) per simulare il router perimetrale (**Router**) e l'altro client (**Client2**);

Entrambe le distro dispongono, già in modalità Live, di tutti gli strumenti necessari ai nostri esperimenti, oltre a segnalarsi per un consumo di RAM alquanto morigerato:

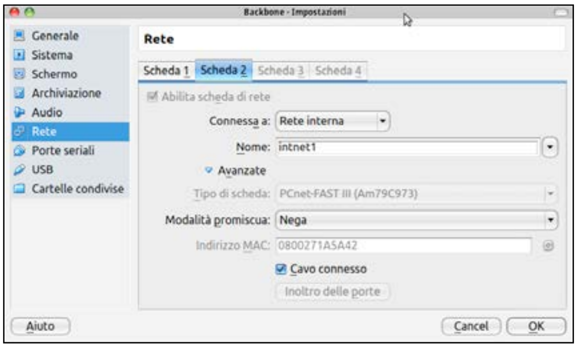
» **Damn Small Linux**, nata per operare su sistemi decisamente datati, in grado di offrire una piena usabilità in appena 64 MB di memoria volatile;

» **Xubuntu 14.04 LTS**, a fronte di una quantità di RAM decisamente superiore seppur non proibitiva per i moderni computer (512 MB), garantisce la disponibilità di tool di sicurezza indispensabili nel proseguo della serie.

Una volta stabilite le distribuzioni da utilizzare, è possibile provvedere alla creazione delle macchine virtuali, iniziando da quella denominata **Backbone** nello schema in **Fig 1**. Dalla schermata principale di Virtualbox, facciamo click sul pulsante **Nuova** per avviare il processo di creazione, quindi inseriamo, nella finestra dialogo che comparirà, il nome della macchina virtuale (Router), il sistema operativo (Linux) e la relativa versione (Linux 2.4 a 32 bit). Cliccando sul pulsante **Avanti** possiamo passare alla schermata di dimensionamento della RAM da assegnare alla macchina virtuale: come premesso, in questo caso sono sufficienti 64 MB. Il passo successivo è rappresentato dalla configurazione del disco fisso: poiché abbiamo stabilito di utilizzare delle distribuzioni Live, non necessitiamo di alcun hard disk, e pertanto possiamo scegliere l'opzione **Non aggiungere un disco fisso virtuale**. La pressione del tasto **Crea** comporta il completamento del processo di creazione della macchina virtuale, previa visualizzazione di un messaggio di avviso da parte di Virtualbox. L'hypervisor si premura di evidenziare come sia impossibile installare un sistema operativo su una macchina virtuale sprovvista di hard disk: poiché, tuttavia, il nostro intento è proprio quello di ricorrere a distribuzioni Live, possiamo tranquillamente ignorare il messaggio, premendo il pulsante **Continua**.
A questo punto la macchina virtuale è pronta per essere utilizzata, ma per consentire il boot della distribuzione da CD (o meglio, dall'ISO che avremo provveduto a scaricare seguendo i link precedentemente forniti) è necessaria qualche piccola configurazione addizionale. Selezionata, dalla finestra principale di Virtualbox, la macchina virtuale, clicchiamo con il tasto destro del mouse e scegliamo la voce **Impostazioni**. Nella finestra così visualizzata, optiamo per la voce **Archiviazione**, quindi clicchiamo sull'icona del CD comparsa accanto all'elemento **Controller: IDE** (**Fig 3**). L'operazione appena compiuta provoca la visualizzazione di un secondo avviso, al quale potremo replicare premendo il pulsante **Scegli il disco**: a questo punto, non ci resta che selezionare l'ISO di DSL e confermare la scelta tramite il pulsante **Apri**. L'ultima operazione da compiere sulla macchina virtuale consiste nella configurazione della scheda di rete: come si evince

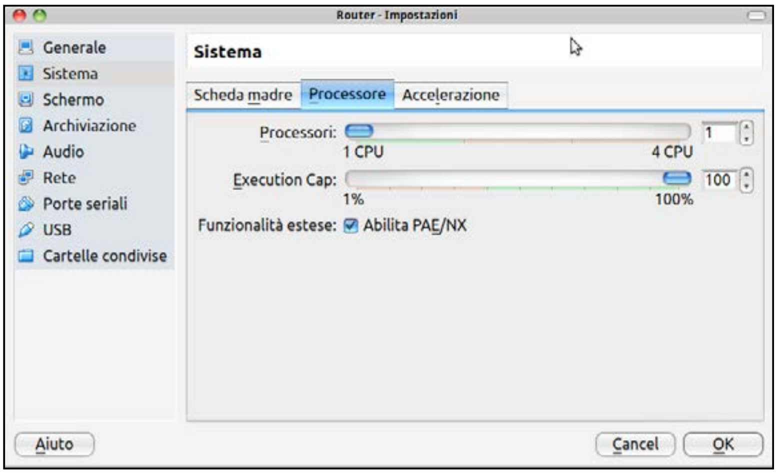


› Fig 4: Ecco la schermata per l'aggiunta di una scheda "solo host" a una VM



› Fig 5: In questa maniera è possibile aggiungere una seconda scheda di rete alla macchina virtuale, consentendole la connessione a una rete interna (*intnet*, nello specifico)

dallo schema in **Fig 1**, la macchina **Backbone** dispone di due interfacce di rete, una collegata alla macchina attaccante, l'altra al router perimetrale della nostra rete SOHO. Ciò non costituisce un problema per Virtualbox, la cui GUI è in grado di gestire l'assegnazione di ben 4 interfacce di rete per macchina virtuale, ciascuna indipendente dalle altre. Per accedere alla schermata di configurazione delle schede di rete, è sufficiente aprire la finestra relativa alle impostazioni della macchina virtuale e selezionare la voce **Rete**. A questo punto, nella parte destra della finestra di dialogo, apparirà la schermata di configurazione delle interfacce di rete, suddivisa in schede (**Fig 4**): alla creazione, Virtualbox fornisce a ciascuna macchina virtuale una singola interfaccia di rete abilitata, denominata **Scheda 1**, demandando all'utente l'eventuale attivazione delle ulteriori schede. A **Scheda 1** affidiamo il compito di garantirci la connettività con la macchina denominata **Attaccante**: poiché, come preannunciato, tale ruolo sarà svolto dalla macchina fisica, ciò richiede l'attestazione della scheda alla rete **solo host** creata nel paragrafo precedente. A tal fine, occorre selezionare, dal menu a tendina denominato **Connessa a**, la modalità **Scheda solo host**, per poi verificare che il nome comparso nella zona sottostante coincida con **vboxnet0**.
La configurazione della seconda scheda si discosta solo leggermente dalla prima: in questo caso, infatti, dobbiamo provvedere ad attivare l'interfaccia, e solo successivamente a disporne il collegamento alla rete interna. Le operazioni da compiere sono le seguenti (**Fig 5**):
» selezionare la scheda denominata **Scheda 2**;
» cliccare sulla checkbox **Abilita la scheda di rete**;
» selezionare, dal menu a tendina denominato **Connessa a:**, la modalità **Rete interna**;



› Fig 6: In alcune architetture, potrebbe essere necessario abilitare la funzionalità *PAE/NX* per garantire il corretto funzionamento delle VM basate su Xubuntu

» verificare che nel campo **Nome:** sottostante compaia la stringa **intnet1**;
» premere il pulsante **OK** per confermare le operazioni effettuate.
Terminato il lavoro preparatorio sulla macchina **Backbone**, passiamo alla macchina **Router**, che andrà configurata come di seguito:
» nome macchina virtuale: **Router**;
» sistema operativo: Linux (Ubuntu a 64 bit);
» RAM: 512 MB;
» disco fisso: nessuno;
» archiviazione: selezionare l'ISO di Xubuntu, secondo la medesima procedura adottata per la macchina **Backbone**;
» Rete: due schede di rete, la prima connessa alla rete interna di nome **intnet1**, la seconda alla rete interna di nome **intnet**.
Al contrario delle macchine basate sulla distribuzione Damn Small Linux, le Virtual Machine Xubuntu, per poter funzionare correttamente, potrebbero richiedere l'abilitazione della funzionalità PAE/NX. Si tratta di estensioni del processore che Virtualbox è in grado di emulare (a patto che siano supportate dal processore della macchina fisica) e che, su alcune piattaforme hardware, potrebbero risultare necessarie alla corretta esecuzione delle VM basate su Ubuntu. Qualora al boot tali VM dovessero presentare tempi di caricamento insolitamente lunghi, è possibile abilitare le funzionalità PAE/NX (**Fig 6**) attraverso la seguente procedura:
» accedere alle impostazioni della macchina virtuale; »

Indirizzi IP pubblici e indirizzi privati

Gli indirizzi IP attualmente in uso (Ipv4) hanno una lunghezza di 4 byte, come appare evidente dalla rappresentazione più diffusa (detta *digital dotted notation*), che ne consente la visualizzazione come 4 gruppi di numeri separati da un punto. Il numero massimo di indirizzi IP esprimibile con 4 byte è, pertanto, di circa 4 miliardi (2^32, un numero nell'ordine di 10^9). Non tutti, tuttavia, sono utilizzabili in Internet: in altre parole, non tutti sono indirizzi IP "pubblici". Alle classi di indirizzi che identificano il cosiddetto localhost (la macchina locale) e a quelle dedicate a broadcast e multicast, si aggiungono infatti i cosiddetti "indirizzi privati", che il documento **Request For Comment** (RFC) 1918 fissa in tre blocchi:
» un blocco compreso tra gli indirizzi 10.0.0.0 e 10.255.255.255;
» un secondo blocco che va dall'indirizzo 172.16.0.0. all'indirizzo 172.31.255.255;
» un terzo blocco compreso tra 192.168.0.0 e 192.168.255.255.
Questi indirizzi hanno la peculiarità di non essere ruotabili su Internet, ma possono essere utilizzati liberamente all'interno di reti private (es. LAN): in altri termini, nessun indirizzo privato è visibile all'esterno della rete privata di appartenenza, e ciò consente a milioni di reti private di condividere il medesimo indirizzamento senza alcuna interferenza reciproca.

```
[/home/dsl]# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
[/home/dsl]# route add -net 212.100.1.0/24 gw 211.100.1.2
[/home/dsl]#
```

› Fig 7: Le configurazioni di rete che consentono alla VM *Backbone* di instradare il traffico da e verso la rete locale oggetto dei nostri test

» selezionare la voce **Sistema**;
» selezionare la scheda **Processore**;
» cliccare sulla checkbox **Abilita PAE/NX**, sulla quale deve essere apposto un flag.

La seconda macchina virtuale basata su Xubuntu è quella denominata **ClientA** nello schema in **Fig 1**, la cui configurazione ricalca quella appena vista:

» nome macchina virtuale: **ClientA**;
» sistema operativo: Linux (Ubuntu a 64 bit);
» RAM: 512 MB;
» disco fisso: nessuno;
» archiviazione: selezionare l'ISO di Xubuntu, secondo la medesima procedura adottata per la macchina **Backbone**;
» Rete: una scheda di rete, connessa alla rete interna di nome **intnet**.

L'ultima VM, invece, è basata su Damn Small Linux, e va configurata come segue:

» nome macchina virtuale: **ClientB**;
» sistema operativo: Linux (Linux 2.4 a 32 bit);
» RAM: 64 MB;
» disco fisso: nessuno;
» archiviazione: selezionare l'ISO di Damn Small Linux, secondo la medesima procedura adottata per la macchina **Backbone**;
» Rete: una scheda di rete, connessa alla rete interna di nome **intnet**.

Configurazione di rete

A questo punto, il nostro laboratorio è pronto per esercitare: se lanciassimo in esecuzione tutte le VM, tuttavia, non otterremmo alcun risultato degno di nota. Aver posto le macchine virtuali contigue sulla medesima subnet, infatti, non comporta automaticamente la garanzia che queste siano in grado di dialogare tra loro. A tal fine, è necessaria un'opportuna configurazione che preveda, per ciascuna macchina, l'assegnazione degli indirizzi IP a ogni interfaccia di rete, nonché l'impostazione delle necessarie rotte statiche. Iniziamo queste operazioni dalla VM denominata **Backbone** che, una volta selezionata all'interno della finestra principale di Virtualbox, può essere avviata semplicemente mediante la pressione del tasto **Avvia**. Ciò provocherà la visualizzazione di una finestra dedicata alla macchina virtuale: per consentire a quest'ultima di assumere il controllo di mouse e tastiera, provvediamo a cliccare sulla suddetta finestra, ignorando eventuali messaggi di avvertimento visualizzati da Virtualbox. È infatti possibile restituire in qualsiasi momento mouse e tastiera alla macchina fisica, mediante la pressione del tasto **CTRL** destro. Il boot della distribuzione Damn Small Linux è subordinato alla visualizzazione di una schermata per la scelta delle modalità di caricamento del sistema operativo: per i nostri scopi, le opzioni di boot di default risultano più che sufficienti, motivo per il quale è possibile superare la schermata con la pressione del tasto **Invio**. In pochi attimi verrà visualizzato il desktop essenziale della distro.
La configurazione dei parametri di rete della macchina

```
root@ubuntu:/home/garrick# ping 210.100.1.2 -c 5
PING 210.100.1.2 (210.100.1.2) 56(84) bytes of data.
64 bytes from 210.100.1.2: icmp_seq=1 ttl=64 time=2.03 ms
64 bytes from 210.100.1.2: icmp_seq=2 ttl=64 time=0.563 ms
64 bytes from 210.100.1.2: icmp_seq=3 ttl=64 time=0.629 ms
64 bytes from 210.100.1.2: icmp_seq=4 ttl=64 time=0.650 ms
64 bytes from 210.100.1.2: icmp_seq=5 ttl=64 time=0.921 ms

--- 210.100.1.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
rtt min/avg/max/mdev = 0.563/0.958/2.031/0.551 ms
root@ubuntu:/home/garrick#
```

› Fig 8: I risultati del ping ci confermano senza ombra di dubbio la raggiungibilità della macchina *Backbone* a partire dalla macchina fisica

virtuale richiede una console con diritti di root, avviabile con la seguente procedura:

» click, con il pulsante sinistro, su un qualsiasi punto del desktop;
» selezione delle voci **Xshell | root access | transparent** del menu a tendina così ottenuto.

Dal terminale visualizzato, provvediamo ad abilitare la prima scheda di rete (quella alla quale è affidata la connessione con la macchina fisica) mediante il comando:

```
#ifconfig eth0 up
```

per poi assegnarle l'indirizzo IP e la relativa netmask, attraverso il comando

```
#ifconfig eth0 210.100.1.2 255.255.255.0
```

La seconda scheda di rete (che assicura il collegamento alla VM denominata **Router**) richiede le medesime operazioni:

```
#ifconfig eth1 up
#ifconfig eth1 211.100.1.1 255.255.255.0
```

la cui esecuzione, sebbene garantisca una piena conformità delle impostazioni di rete con quanto descritto nello schema in **Fig 1**, non consente ancora ai pacchetti inviati da **Attacker** di raggiungere i due client, e viceversa. Perché ciò sia possibile, infatti, è necessario (**Fig 7**):

» abilitare il routing sulla macchina:

```
#sysctl -w net.ipv4.ip_forward=1
```

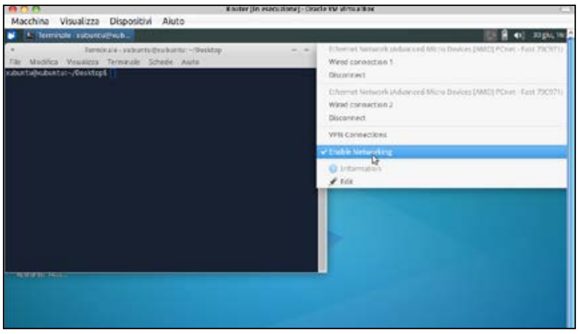
» impostare la rotta che consente il raggiungimento dei due client per mezzo della VM **Router**:

```
#route add -net 212.100.1.0/24 gw 211.100.1.2
```

Possiamo ottenere una prima, rapida, conferma in merito alla bontà delle operazioni appena compiute provando a configurare anche la macchina fisica. Dopo aver aperto un terminale, digitiamo (da root) i comandi:

```
#ifconfig vboxnet0 210.100.1.1 255.255.255.0
# route add default gw 210.100.1.2
```

quindi eseguiamo il comando **ping** per verificare



› Fig 9: Per disabilitare la funzionalità di configurazione automatica delle reti (non necessaria ai nostri scopi) è sufficiente un semplice click

l'operatività del collegamento tra la macchina fisica e la VM **Backbone** (**Fig 8**):

```
# ping -c 5 210.100.1.2
```

Terminata la configurazione dei primi due "computer" dello schema in **Fig 1**, possiamo passare alla macchina denominata **Router** che, come abbiamo visto nel paragrafo precedente, è simulata per mezzo di una VM basata su Xubuntu. Premendo il tasto **Enter** nella finestra della macchina virtuale subito dopo averla avviata è possibile visualizzare la schermata grafica di selezione della lingua (italiano) e delle modalità operative della distribuzione. La scelta di nostro interesse è costituita dalla voce **Prova Xubuntu senza installarlo**: dopo aver confermato la selezione mediante il tasto **Invio**, sarà caricata la distribuzione, e visualizzato il relativo desktop. La configurazione dei parametri di rete richiede, come operazione preliminare, la disabilitazione delle funzionalità di networking assistito, offerta dal plug-in **Network connections** del pannello orizzontale visualizzato in cima al desktop. Tale operazione può essere effettuata cliccando sull'icona del plug-in (**Fig 9**), e provvedendo a deselezionare la voce **Enable Networking** dal menu così visualizzato. A questo punto è possibile aprire un terminale, cliccando con il tasto destro del mouse in un qualsiasi punto della scrivania e scegliendo la voce **Apri un terminale qui**. All'interno del terminale digitiamo i seguenti comandi:

```
#sudo ifconfig eth0 211.100.1.2 255.255.255.0
#sudo ifconfig eth1 212.100.1.1 255.255.255.0
#sudo sysctl -w net.ipv4.ip_forward=1
#sudo route add -net 210.100.1.0/24 gw 211.100.1.1
```

Reiterando le operazioni sin qui analizzate, possiamo procedere alla configurazione delle schede di rete dei due client (**ClientA**, basato su Xubuntu, e **ClientB**, con distribuzione Damn Small Linux). Coerentemente con quanto riportato in **Fig 1**, per **ClientA** sono sufficienti i comandi:

```
#sudo ifconfig eth0 212.100.1.2 255.255.255.0
#sudo route add default gw 212.100.1.1
```

mentre per **ClientB** occorre digitare:

```
#ifconfig eth0 up
#ifconfig eth0  212.100.1.3 255.255.255.0
# route add default gw 212.100.1.1
```

I rischi dell'esposizione

Abbiamo riprodotto perfettamente lo scenario descritto a inizio articolo (cit. **Fig 1**): le due macchine client, **ClientA** e **ClientB**, sono esposte su **Internet** senza protezione alcuna. Ciò si traduce nella possibilità, per un generico attaccante quale la macchina **Attacker**, di contattare i due client. Gli esiti del comando **ping**, da eseguire sulla macchina fisica (che, come già detto, simula la macchina **Attacker**), possono confermare quanto appena affermato:

```
#ping -c 5 212.100.1.2
#ping -c 5 212.100.1.3
```

Come mostrato dalla **Fig 10**, tutti i pacchetti ICMP inviati dall'attaccante verso i due client hanno determinato una risposta da parte di questi ultimi. Un utente inesperto potrebbe pensare che rispondere a un ping proveniente da Internet non costituisca un problema così grave... per confutare questa tesi, chiediamoci allora: cosa succederebbe se, a causa di una configurazione troppo permissiva, i client presentassero anche delle porte aperte? Per rispondere a questa domanda, provvediamo ad avviare alcuni servizi sulle due workstation, iniziando da **Client2**. Damn Small Linux, infatti, dispone già in modalità Live di ben 3 servizi di uso comune: un server Web, un demone SSH e un server FTP che, tuttavia, non risultano attivi all'avvio della distribuzione. Possiamo ovviare a questo inconveniente facendo click con il pulsante sinistro in qualsiasi punto del desktop di **ClientB**, per poi selezionare, nei menu a tendina così visualizzati, le voci:

- **System | Daemons | Monkey Web Server | monkey start** per avviare il server Web;
- **System | Daemons | ssh | start** per il demone SSH;
- **System | Daemons | FTPd | betaftpd start** per il server FTP.

Per quanto possa sembrare strano, Xubuntu non dispone, in modalità Live, di un tale assortimento di servizi pronti all'uso. Questa limitazione, tuttavia, può essere superata grazie al tool **netcat** (nc), che possiamo adoperare per simulare un arbitrario servizio di rete. A tal fine, sfrutturemo una delle tante funzionalità di questo fantastico software, definito non a caso il "coltellino svizzero" dei tool di networking: la possibilità di porsi in ascolto su una qualsiasi porta di rete. Possiamo dunque ottenere il nostro scopo provvedendo ad aprire un terminale su ClientA e digitando il comando:

```
#sudo nc -l -p 8080
```

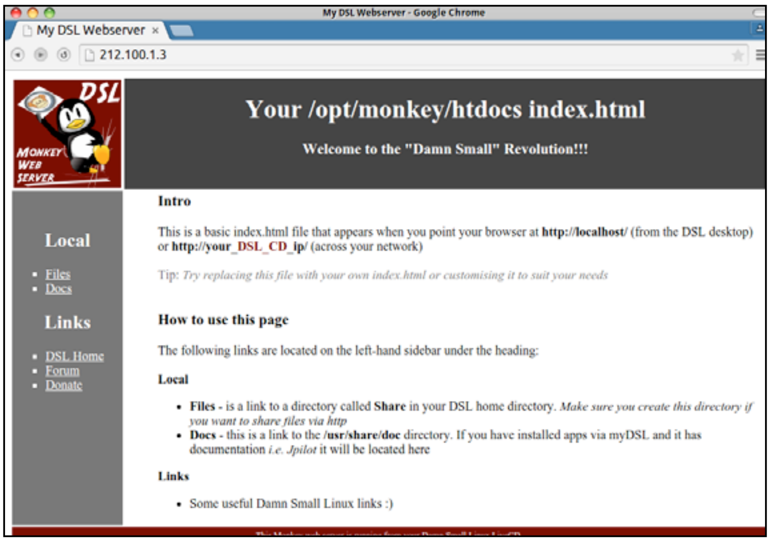
Al contrario di quanto accade per ClientB, il server Web simulato sarà in ascolto sulla porta 8080, tradizionalmente associata (insieme alla più nota 80) proprio ai servizi HTTP. Completata anche questa operazione, disponiamo dunque di servizi in esecuzione su entrambi i client, da noi esplicitamente avviati. Tuttavia, è bene non ingannarsi su questo punto: spesso infatti (soprattutto in alcuni sistemi operativi commerciali) le workstation offrono servizi di cui l'utente è del tutto ignaro. Tali servizi, in uno scenario come quello che stiamo analizzando, sono soggetti ai medesimi rischi di sicurezza che andremo a descrivere nel proseguo: ecco un buon motivo per conoscere nel dettaglio la »

```
garrick@ubuntu:~$ ping -c 5 212.100.1.2
PING 212.100.1.2 (212.100.1.2) 56(84) bytes of data.
64 bytes from 212.100.1.2: icmp_seq=1 ttl=62 time=4.20 ms
64 bytes from 212.100.1.2: icmp_seq=2 ttl=62 time=3.12 ms
64 bytes from 212.100.1.2: icmp_seq=3 ttl=62 time=2.94 ms
64 bytes from 212.100.1.2: icmp_seq=4 ttl=62 time=2.49 ms
64 bytes from 212.100.1.2: icmp_seq=5 ttl=62 time=2.92 ms

--- 212.100.1.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 2.490/3.137/4.202/0.574 ms
garrick@ubuntu:~$ ping -c 5 212.100.1.3
PING 212.100.1.3 (212.100.1.3) 56(84) bytes of data.
64 bytes from 212.100.1.3: icmp_seq=1 ttl=62 time=5.00 ms
64 bytes from 212.100.1.3: icmp_seq=2 ttl=62 time=2.47 ms
64 bytes from 212.100.1.3: icmp_seq=3 ttl=62 time=2.76 ms
64 bytes from 212.100.1.3: icmp_seq=4 ttl=62 time=2.67 ms
64 bytes from 212.100.1.3: icmp_seq=5 ttl=62 time=2.74 ms

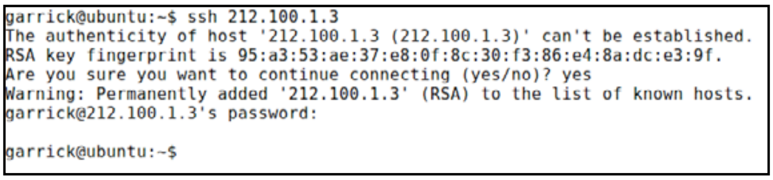
--- 212.100.1.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 2.474/3.133/5.007/0.943 ms
garrick@ubuntu:~$
```

› Fig 10: La raggiungibilità dei due client della rete dalla VM *Attacker* è pienamente confermata dai risultati dei ping

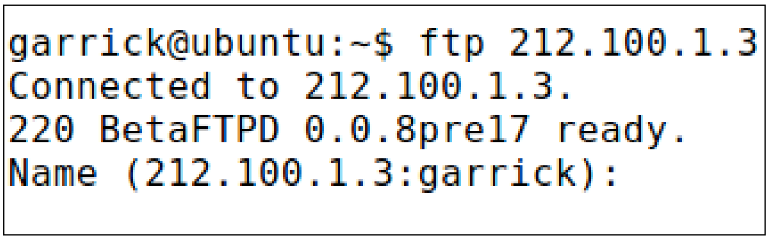


› Fig 11: Di fronte a una schermata del genere, non si può che essere certi della piena fruibilità del server Web dalla macchina *Attacker*

configurazione del proprio PC! Inoltre, che siano stati mandati in esecuzione manualmente dall'utente, o che siano avviati automaticamente al boot del sistema operativo, i servizi offerti da una workstation non sono, tipicamente, pensati per essere utilizzati al di fuori della workstation stessa (o, al più, della rete locale in cui questa si trova). Una configurazione di rete mal realizzata, tuttavia, consente di allargare a dismisura la platea dei possibili fruitori del servizio, incrementando di conseguenza la possibilità che tra essi siano inclusi utenti malevoli. Cosa potrebbe accadere, per esempio, se il mio server FTP, utilizzato per la condivisione di file multimediali tra i PC della mia rete casalinga e dotato di una configurazione minimale, fosse esposto involontariamente a Internet? Possiamo rispondere immediatamente a questa domanda analizzando le opzioni disponibili alla macchina **Attacker** a seguito dell'avvio dei servizi su **ClientA** e **ClientB**. Dagli esperimenti precedenti con il comando ping, abbiamo appreso come **Attacker** sia in grado di raggiungere entrambi i client: ciò significa che, in assenza di dispositivi di filtraggio a protezione delle workstation, ciascun servizio in esecuzione su queste ultime sarà fruibile anche ad **Attacker**! Per esempio, se prendiamo in considerazione



› Fig 12: Anche il servizio SSH è esposto alla macchina *Attacker*



› Fig 13: Il server FTP, in esecuzione sul client basato su DSL, risponde senza problema alle interrogazioni provenienti dalla macchina *Attacker*

ClientB, la macchina malevola è in grado di:

» interrogare il server Web mediante ricorso a un semplice browser (come mostrato in **Fig 11**, è sufficiente digitare nella barra degli indirizzi l'URL **http://212.100.1.3**, per visualizzare la pagina di default del server Web Monkey);

» collegarsi in ssh alla workstation, mediante il comando seguente (**Fig 12**):

```
# ssh 212.100.1.3
```

» collegarsi in ftp alla workstation, mediante il comando seguente (**Fig 13**):

```
# ftp 212.100.1.3
```

Lo stesso vale per il server Web simulato su ClientA, che può essere contattato dalla macchina attaccante mediante ricorso al browser (la stringa da digitare nella barra degli indirizzi è **http://212.100.1.2:8080**): in questo caso, tuttavia, il nostro browser non otterrà risposta alcuna, in quanto l'istanza di netcat in esecuzione su ClientA si limiterà a ricevere e visualizzare a video la richiesta HTTP generata da **Attacker**, rimanendo in attesa dell'eventuale input dell'utente (**Fig 14**). Appurato che la raggiungibilità da Internet dei nostri client si traduce nella possibilità, per un attaccante, di accedere a eventuali servizi in esecuzione su di essi, esaminiamo le potenziali conseguenze di questa situazione. Anche limitandoci ai soli servizi in esecuzione su ClientA e ClientB, non possiamo esimerci dall'osservare come la loro esposizione su Internet comporti gravi rischi alla sicurezza delle workstation e dell'intera LAN: l'assenza di un qualsiasi filtro dinanzi ai client consente, a un attaccante, il pieno accesso a funzionalità pensate (e, soprattutto, configurate) per una fruizione nel ristretto perimetro della rete locale. Nella realtà quotidiana, quindi, consentire l'accesso da Internet a servizi quali SSH, FTP e HTTP, non opportunamente protetti, comporta un'elevata probabilità di divenire oggetto di "attenzioni indesiderate", che potrebbero tradursi in vere e proprie intrusioni. Come? Vediamo di seguito qualche possibile scenario...

HTTP

L'accessibilità, da parte di un attaccante, a un server Web non correttamente configurato, può comportare la possibilità di consultarne tutti i contenuti. Un tipico esempio riguarda i server che supportano la funzionalità di directory listing: in assenza di un'apposita pagina iniziale, tale funzionalità impone al server di elencare ogni file e sottodirectory da questo gestita, consentendone la visualizzazione da browser. Qualora le possibili ricadute sulla vostra privacy non vi sembrassero abbastanza, possiamo soffermarci a ragionare sulle conseguenze dell'esposizione di eventuali pagine dinamiche ivi presenti, meglio ancora se supportate da un database di backend. Per esempio, agli utenti più curiosi capita spesso di installare software nuovi per provarne le caratteristiche: se questi software sono *Content Management System* (CMS) Open Source come Joomla, o Wordpress, la naturale sede di installazione è costituita proprio da un server Web. Capita altrettanto spesso, purtroppo, che i medesimi utenti spostino la propria attenzione su un software nuovo, abbandonando quello precedente: ma se questo software è un CMS e non si adottano le dovute precauzioni, ciò può tradursi nella presenza sul sistema di un server Web attivo al boot, dal quale è possibile consultare il CMS in questione. Con il

passar del tempo, saranno scoperte numerose vulnerabilità per la versione del CMS "dimenticata" sul server: vulnerabilità che, a seconda della gravità, potrebbero portare addirittura alla compromissione dell'intera macchina.

FTP

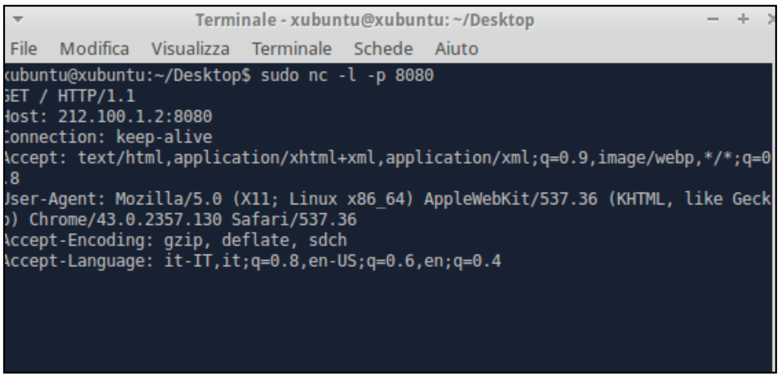
L'accessibilità, da parte di un attaccante, a un server Web non correttamente configurato, può comportare la possibilità di consultarne tutti i contenuti. Un tipico esempio riguarda i server che supportano la funzionalità di directory listing: in assenza di un'apposita pagina iniziale, tale funzionalità impone al server di elencare ogni file e sottodirectory da questo gestita, consentendone la visualizzazione da browser. Qualora le possibili ricadute sulla vostra privacy non vi sembrassero abbastanza, possiamo soffermarci a ragionare sulle conseguenze dell'esposizione di eventuali pagine dinamiche ivi presenti, meglio ancora se supportate da un database di backend. Per esempio, agli utenti più curiosi capita spesso di installare software nuovi per provarne le caratteristiche: se questi software sono *Content Management System* (CMS) Open Source come Joomla, o Wordpress, la naturale sede di installazione è costituita proprio da un server Web. Capita altrettanto spesso, purtroppo, che i medesimi utenti spostino la propria attenzione su un software nuovo, abbandonando quello precedente: ma se questo software è un CMS e non si adottano le dovute precauzioni, ciò può tradursi nella presenza sul sistema di un server Web attivo al boot, dal quale è possibile consultare il CMS in questione. Con il passar del tempo, saranno scoperte numerose vulnerabilità per la versione del CMS "dimenticata" sul server: vulnerabilità che, a seconda della gravità, potrebbero portare addirittura alla compromissione dell'intera macchina.

FTP

In reti SOHO, il *File Transfer Protocol* (FTP) è utilizzato, generalmente, per la condivisione di file tra gli utenti. In una rete domestica, per esempio, un server FTP con accesso anonimo (ovvero che non richieda l'inserimento di alcuna credenziale per l'accesso) può costituire un modo rapido e pratico per consentire a più dispositivi, magari dotati di sistemi operativi differenti (Windows, Linux, Android, ecc.), di accedere ai medesimi file. In un tale scenario l'assenza di autenticazione per le operazioni di lettura e scrittura su server può essere vista addirittura come un valore aggiunto, in quanto rende più semplici e veloci le attività di condivisione. Le cose cambiano radicalmente se invece il server risulta accessibile da Internet: immaginate cosa potrebbe capitare se qualche malintenzionato sfruttasse l'assenza di credenziali per cancellare tutti i documenti conservati sul server! Non solo: un attaccante particolarmente diabolico potrebbe decidere di iniettare nei file ospitati sul server del codice virale, al fine di prendere il controllo di tutta la rete... O peggio, potrebbe optare per lo sfruttamento dello spazio da voi gentilmente offerto per le proprie attività illecite o, ancora, condividere con altri utenti del Web, a voi sconosciuti, materiali compromettenti sotto la vostra responsabilità...

SSH

Il protocollo SSH costituisce uno strumento inestimabile per la configurazione remota di macchine GNU/Linux.



› Fig 14: Anche il servizio simulato tramite netcat dal client basato su Xubuntu è pienamente raggiungibile dalla macchina *Attacker*, come mostra la richiesta HTTP inviata dal browser di quest'ultima sulla porta 8080

Pensato principalmente per i server, può risultare di grande utilità anche nell'uso quotidiano: per esempio un server SSH, installabile nelle moderne distribuzioni con un singolo comando, può consentire a un amico fidato di intervenire da remoto sulla vostra Linux Box per risolvere un problema di configurazione. In uno scenario come quello appena descritto, all'ipotetico amico andrebbero fornite le credenziali di autenticazione (username e password) di un utente locale o addirittura di un amministratore della macchina: cosa può accadere se tali credenziali, magari create ad hoc per l'esigenza, risultano essere poco robuste? Nel caso in cui si "dimentichi" il server SSH attivo anche a cessata emergenza, è possibile che qualche malintenzionato possa provare ad accedervi pur non possedendo lecitamente le credenziali utente richieste al login. In uno scenario del genere, la presenza sul sistema di utenti dotati di credenziali poco robuste (per esempio un tipico utente di test, con username pari a **user** e password pari a **password**) potrebbe, potenzialmente, compromettere la sicurezza dell'intera macchina.

Tirando le somme...

Le simulazioni condotte in queste pagine ci portano a una conclusione comune: uno dei modi per assicurare la sicurezza della nostra Linux box consiste nel verificarne con scrupolosità la configurazione avendo cura, in particolare, di eliminare gli eventuali servizi non essenziali. Quella che può sembrare una buona soluzione sul piano teorico, tuttavia, si scontra con le necessità correlate con l'uso pratico delle nostre macchine. I servizi esaminati, infatti, sono in grado di risolvere in maniera semplice alcune esigenze comuni, e pertanto potrebbero risultare necessari anche nell'uso quotidiano. In tal caso, non potendo optare per la disabilitazione, è necessario garantirne l'esecuzione in una cornice di sicurezza, che non può prescindere da un'attenta configurazione. Cercare in Rete una guida per l'hardenizzazione del servizio da noi richiesto può senz'altro costituire un buon inizio: ma accanto alle soluzioni specifiche per ciascun software, è possibile ricorrere a strumenti di uso più generale, in grado di migliorare il livello di sicurezza dell'intera Linux Box e, addirittura, dell'intera rete locale. Quali sono, allora, questi strumenti, e come si utilizzano? Per scoprirlo, non vi resta che seguire la prossima puntata! LXP

Controllare gli accessi di root

Lo staff di Linux Pro vi svelerà come ottenere i poteri divini del superuser, grazie a **sudo**, al fine di esaudire i desideri degli esseri inferiori



Linux tiene gli utenti amministrativi separati da tutti gli altri utenti del sistema. Anche se siete l'unico utente sul computer, esiste comunque un account root o superuser che dovete utilizzare per fare certe cose. Alcune distribuzioni, come Ubuntu, cercano di nascondere l'account del superuser: non potete fare login come root ma comunque esiste e il sistema non funzionerebbe senza di lui. Ma perché a volte è necessario lavorare come root? Come farlo e come si può fare per permettere ad altri utenti di compiere attività come root offrendogli il controllo completo del sistema?
Il modo tradizionale per dare accesso come root è quello di utilizzare il comando su. Questo comando che sta per switch user (cambia utente), e non per super user, vi permette di diventare un altro utente per tutto il tempo necessario. Quando viene eseguito senza argomenti da una shell, vi consente di diventare l'utente root (purché ne conosciate la password). Non è esattamente come fare login con l'utente root, alcuni elementi dell'ambiente rimangono i vostri. Per avvicinarsi il più possibile all'ambiente reale di login possiamo utilizzare l'opzione -l. Abbiamo detto che su permette di cambiare utente e che di default ci porta all'utente root, ma potete passare uno username al comando, per esempio:

```
$ su -l paolo
```

Una volta eseguito su, diventerete quell'utente finché non scriverete **exit** o premerete **Ctrl**+**D**. Questa non è una cosa buona, lasciare una shell aperta come root non è molto sicuro, per questo esiste anche la possibilità di passargli un unico comando utilizzando l'opzione -c

```
$ su -c "ifconfig eth0 up"
```

Questo comando non apre una sessione shell, invece, dopo aver chiesto la password, esegue il comando dato.

E poi arriva sudo

Il comando su ha alcuni inconvenienti: richiede che l'utente conosca la password di root, inoltre permette agli utenti di fare ciò che vogliono e di lasciare una shell di root in esecuzione quando hanno finito di fare quello che dovevano fare. Il metodo su -c è lento se volete eseguire più di un comando, dovete inserire la password di root ogni volta. La risposta è sudo, supportato ormai dalla maggior parte delle distribuzioni.
Il comando sudo funziona diversamente. Per prima cosa, è utilizzato per eseguire un singolo comando, per esempio:

```
$ sudo ifconfig eth0 up
```

Da notare che non c'è bisogno di includere il comando tra virgolette come prima. Inoltre, vi chiede la vostra password e non quella del superuser, quindi non c'è la necessità di dare a qualcuno la password di root (in realtà di default l'utente root di Ubuntu non ce l'ha proprio). Provando a eseguire un altro comando con sudo noterete un'altra differenza, non vi chiede nuovamente la vostra password. Vi permette di eseguire un nuovo comando entro cinque minuti dall'esecuzione dell'ultimo senza necessità di inserire nuovamente la password. Il tempo di timeout può essere modificato tramite file di configurazione se volete. Se l'utente deve fornire soltanto la propria password per fungere da root, cosa impedisce a qualsiasi utente di eseguire comandi come root? La risposta si trova nel file di configurazione ***/etc/sudoers***. Questo file specifica esattamente chi può fare cosa, offrendo un controllo dettagliato sui permessi di esecuzione. La configurazione di default è quella di permettere ai membri del gruppo wheel di eseguire qualunque comando. Alcune distribuzioni utilizzano un nome diverso per il gruppo, come admin (wheel, è il nome tradizionale dato a un gruppo con elevati privilegi già da tempi antecedenti Unix). Molti installer inseriscono l'utente creato in fase di installazione in quel gruppo, gli utenti successivi invece

no, a meno che non li aggiungiate voi. Questo significa che l'utente che ha installato la distribuzione ottiene automaticamente i privilegi di admin, senza che sia necessario che qualcuno apra un terminale come root o anche solo che conosca la password di root.

visudo

Potete dare i privilegi di sudo ad altri utenti modificando il file ***/etc/sudoers***, ma non dovreste mai farlo manualmente. Se fate un'errore di sintassi all'interno di questo file quando lo modificate, potreste far sì che sudo smetta di funzionare. E siccome necessitate di lui per modificare quel file (in realtà dovete essere root anche solo per leggere questo file per motivi di sicurezza) non sareste più in grado di riparare al danno. La risposta è nel comando visudo che crea una copia temporanea del file ***/etc/sudoers*** e la carica in un editor. Dopo che avrete modificato il file, visudo controllerà che sia corretto prima di sovrascrivere il file ***/etc/sudoers*** reale. Il nome farebbe pensare che utilizzi vi per le modifiche, ma non preoccupatevi, è soltanto una possibilità. In effetti, visudo utilizza il vostro editor di default, che potete definire tramite la variabile di ambiente EDITOR, o tramite i tool per la configurazioni di Ubuntu e delle altre distribuzioni basate su Debian. Per cambiare l'editor di default del sistema lanciate:

```
$ sudo update-alternatives --config editor
```

Potete impostare la variabile di ambiente EDITOR nel vostro profilo o impostarla da riga di comando, in questo modo

```
$ EDITOR="gedit" sudo -E visudo
```

Il -E è necessario per assicurarsi che la variabile di ambiente venga passata anche all'utente root, vedremo qualcosa sulle variabili di ambiente in seguito. Ora sappiamo come modificare il file **sudoers**, vediamo cosa possiamo farci. Il file contiene due tipi principali: gli alias, che sostanzialmente sono delle definizioni di variabili, e i dati degli utenti. Questi ultimi determinano chi può fare cosa, per esempio nel file di default troviamo questa riga:

```
%wheel ALL=(ALL) ALL
```

Il formato di questa riga è "chi dove = (come_chi) cosa". Il simbolo % all'inizio del nome significa che è un gruppo e la regola si applica a tutti i membri del gruppo. Il 'dove' è l'hostname del computer e l'oggetto seguente indica con quali utenti lanciare il comando. L'ultimo oggetto è una lista di comandi che possono essere eseguiti. ALL è la parola jolly di sudo, quindi questa riga significa che i membri del gruppo wheel possono eseguire qualsiasi cosa in qualsiasi posto, la riga standard per l'utente amministrativo. Se volete dare a un nuovo utente i privilegi di root il modo più semplice di farlo è quello di aggiungerlo al gruppo wheel (o admin, a seconda della vostra distribuzione):

```
$ sudo gpasswd -a michele wheel
```

Limitare l'accesso di root

Ma cosa succede se vogliamo che Michele non sia in grado di fare qualsiasi cosa, ma possa modificare i file Web di proprietà dell'utente Apache? Possiamo utilizzare qualcosa di simile a

```
michele ALL=(apache) ALL
```

In questo caso non mettiamo il % perché è un utente. Stiamo ancora permettendo tutti i comandi su tutti gli host, ma soltanto come utente Apache

```
$ sudo emacs index.html
```

darà errore, perché sudo, come su di default punta al superuser se non diversamente specificato.

Fare login come root

Anche se l'account di root è disabilitato potete abilitarlo così:

```
$ sudo bash
$ sudo --login
```

Potete aggiungere ! al file **suoders** per vietare l'uso di alcuni comandi:

```
%wheel ALL=(ALL) ALL, !/bin/bash, !/bin/sh, !/bin/zsh
```

Perché dovreste volerlo fare? Una ragione è che ogni comando eseguito con sudo viene loggato, rendendo semplice capire chi ha fatto cosa in un sistema con diversi utenti amministratori. Entrare in una shell di root, come abbiamo visto prima, farebbe perdere le proprie tracce.

```
$ sudo -u apache emacs index.html
```

questo invece funzionerà perché utilizza l'utente corretto. Ovviamente, l'utente Michele può ancora eseguire tutti i comandi non amministrativi, anche se lo spazio in cui può farlo è ristretto. Se vogliamo limitare la sua attività alla modifica dei file, possiamo fare così:

```
michele ALL=(apache) /usr/bin/emacs, /usr/bin/vi
```

Abbiamo sostituito l'ALL finale con un elenco di comandi, utilizzando il percorso completo e separandoli con una virgola. Possiamo anche specificare un insieme di argomenti, così:

```
michele ALL=(ALL) /bin/mount /dev/sd[c-f]*, /bin/umount /dev/sd[c-f]*
```

Questo permetterà a Michele di montare e smontare filesystems, ma non **sda** o **sdb**, quindi può lavorare con dischi esterni. Esiste un'importante aggiunta a questa riga (in realtà ne esistono diverse nella documentazione), l'opzione NOPASSWD che fa sì che l'utente possa utilizzare i comandi senza dover reinserire la propria password, usatela con attenzione!

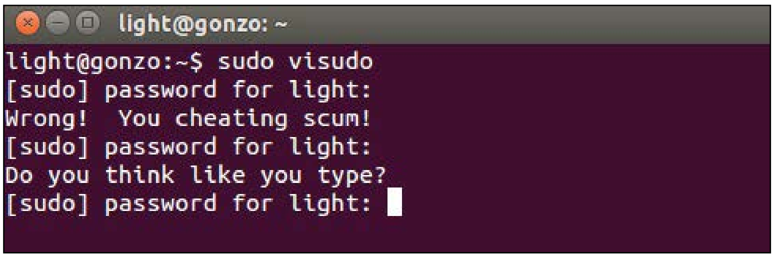
```
michele ALL=(apache) NOPASSWD: /usr/bin/emacs, /usr/bin/vi
```

Variabili di ambiente

Linux utilizza diverse variabili di ambiente per mantenere le informazioni, per esempio $HOME e $DISPLAY. Il comportamento di default di sudo è quello di pulire molte di queste variabili per l'esecuzione del comando. E non sempre è ciò che si vuole. Un modo per aggirare questa cosa è quello di eseguire sudo con l'opzione -E che le mantiene tutte. Una soluzione più strutturata è quella di specificare nel file **sudoers** quali variabili volete mantenere, per esempio potreste avere la necessità di eseguire programmi dotati di GUI con sudo:

```
Defaults env_keep += "DISPLAY HOME"
```

Esistono diverse altre impostazioni che potete aggiungere al file ***/etc/sudoers***, tutte sono documentate, ma ciò che abbiamo visto fin qui dovrebbe essere più che sufficiente per permettervi di iniziare. LXP



› ***Sudo* può essere configurato in modo che insulti chi sbaglia la password. Purtroppo gli insulti sono in inglese**

Chromebook: Costruirsene uno

Ecco come installare **Chrome OS** su di un vecchio laptop e far funzionare le applicazioni di Google

La prima domanda che vi verrà in mente è "perché dovrei voler installare Chrome OS, anche su di un vecchio portatile, quando ci sono moltissime distribuzioni Linux perfettamente funzionanti tra cui scegliere?". Bella domanda: la risposta è che non tutti vogliono una distribuzione completa (e pesante) di Linux, o magari non possono usarla. Parte del successo dei Chromebook, e hanno avuto successo con i loro 5,7 milioni di dispositivi venduti nel 2014 e i 7,3 milioni previsti nel 2015 (fonte: Gartner), è la leggerezza del loro sistema operativo, che è una versione ridotta di Gentoo. Se si vuole dare a qualcuno un accesso facile e chiaro ai servizi di Google è una scelta allettante. La buona notizia è che è facile installare Chromium OS, che è il nome ufficiale del progetto Open Source corrispondente a Google Chrome OS, che purtroppo è disponibile soltanto per i PC Chromebook. Anche se può sembrare diverso con il suo tema blu, Chromium OS offre le stesse funzionalità e servizi, oltre agli stessi vantaggi gestendo tutte le informazioni salvate nel vostro cloud Google. Basato su Gentoo, deriva da Linux e quindi offre tutti i vantaggi del suo kernel, ma tenete presente che è nato soltanto nel 2009 quindi se state pensando di installarlo su hardware antecedente potreste rimanere delusi. In ogni caso noi lo abbiamo provato su di un portatile Lenovo X200 del 2008 e tutto ha funzionato senza alcun problema. In linea generale tutto l'hardware che segue gli standard Intel dovrebbe funzionare senza problemi. Una cosa che sappiamo è che gli adattatori wireless non Intel spesso causano problemi. Diversi siti offrono versioni aggiornate di Chromium OS. È improbabile che vogliate ricompilarlo dai sorgenti, quindi esistono diverse versioni pronte per **VirtualBox** o per eseguirlo e installarlo tramite memorie USB. Noi abbiamo preferito scegliere tra le versioni di **http://arnoldthebat.co.uk** per la loro frequenza di aggiornamento. Un'altra versione molto popolare è disponibile sul sito Web di Hexxeh all'indirizzo **http://chromeos.hexxeh.net**. Non è molto aggiornata, ma funziona.

Un giro su Chromium OS

Esistono molte possibilità per provare Chromium OS. La più semplice è quella di scaricare un'immagine, salvarla su un disco USB o una scheda SD e avviare il portatile o il PC da lì. Esistono anche CD con versioni live, ma avere la possibilità di scrivere sul device permette di salvare anche le impostazioni e portarsele dietro. Non è molto pubblicizzato ma esiste anche un comando che copia con semplicità le partizioni dell'OS sull'hard disk interno. Questa operazione cancella tutti gli altri dati presenti ma con i prezzi sempre più esigui per un piccolo disco SSD non è una cattiva idea usare un disco dedicato. Noi abbiamo provato un dual boot con Chromium e Linux Mint, ma Chromium ha opposto resistenza e si è rifiutato di partire. Fateci sapere se avete più fortuna, ci sembrava una cosa relativamente semplice, creare le due partizioni per ROOT e STATE, fare **dd** dal disco USB su di esse e poi aggiornare **Grub**.

Tip

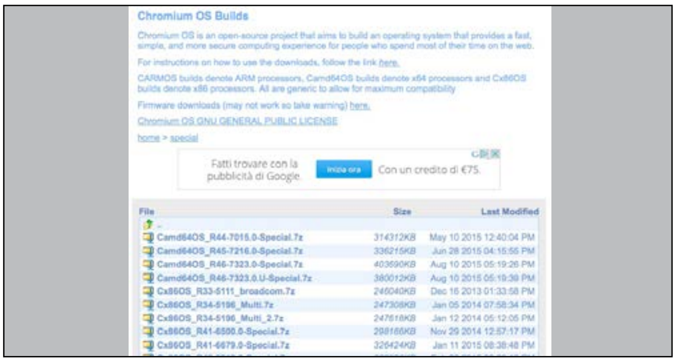
Se utilizzate un dispositivo USB sarete in grado di avviarlo sulla maggior parte dei portatili, netbook e computer desktop. Quindi potete usare Linux ovunque andiate.

Chiavi delle API Google

Questa parte del tutorial non è strettamente necessaria e francamente fastidiosa da implementare perché le chiavi sono molto lunghe e se i vostri font vi fanno confondere i caratteri come la l (L) e la I (i) il tutto diventa ancora più difficile. Ma se volete che la app **Files** funzioni correttamente con il vostro account di Google Drive, è l'unico modo. Dovrete prendere le chiavi da **https://console.developers.google.com**. Sotto API dovete scegliere **Credentials** seguito da **Add credential** -> **OAuth 2.0 client ID** scegliete **Other**, e poi selezionate **Create**. Il secondo passo è quello di aggiungere le variabili di ambiente in Chromium. Con **vi** aggiungete le seguenti tre righe alla fine del file di configurazioni insieme alla nuove chiavi.

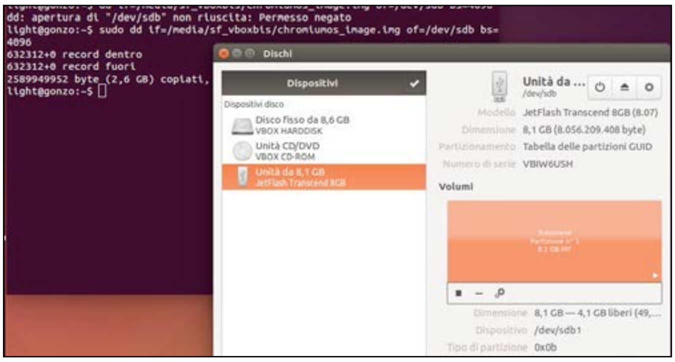
```
GOOGLE_API_KEY=
GOOGLE_DEFAULT_CLIENT_ID=
GOOGLE_DEFAULT_CLIENT_
SECRET=
```

Installare e aggiornare Chromium OS



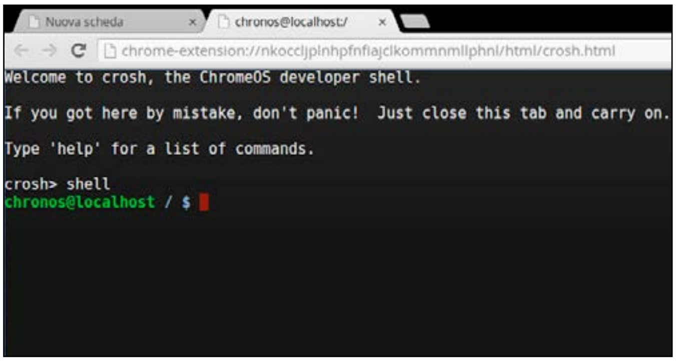
1 Scaricare una versione

Scaricate l'ultima versione speciale dal sito **http://bit.ly/LXFthebat**, qui potete trovare tutte le versioni per i vecchi driver, incluse le schede wireless Broadcom oltre alle versioni non testate per Realtek e Atheros. Notate che alcune sono disponibili sia in versione 32 bit che 64 bit quindi cercate di scaricare la versione più adatta al vostro hardware per sfruttarlo al meglio.



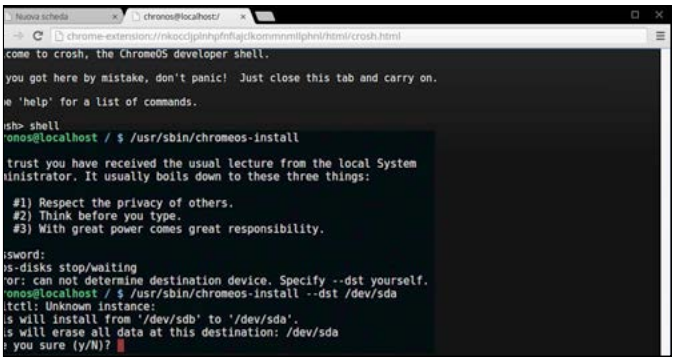
2 Il primo avvio

Il file 7ZIP dovrebbe essere circa 3/400 MB, l'immagine scompattata sarà di circa 2,6 GB. Ora bisogna prendere questa immagine e fare **dd** su un disco USB o una scheda SD sufficientemente grande. Potrete eseguire e usare Chromium OS da qui, ma ovviamente per motivi di velocità installarla su un SSD interno o un altro HDD è l'opzione migliore: $ sudo dd if=<ImmagineChrome.img>of=7dev/sdbXbs=4096.



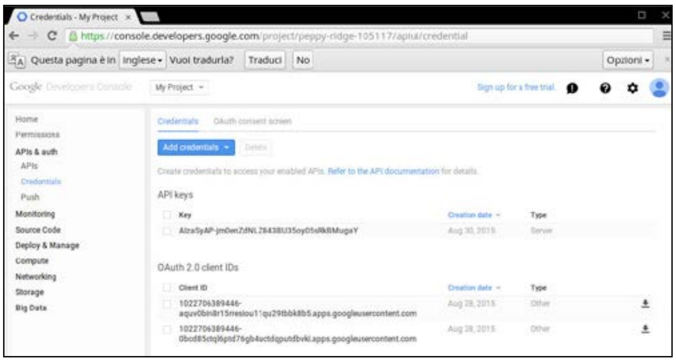
3 Avviare Chromium OS

Avviare il portatile con la penna USB inserita. Probabilmente dovrete premere **F12/Del** o qualunque altro tasto per entrare nelle proprietà del BIOS e avviare da USB. Non sarà necessario fare login con un account dato che si può entrare come ospiti. Superato il wizard iniziale, premete **Ctrl+ALT+T** e si aprirà la Shell di sviluppo di Chrome chiamata crosh. Scrivendo shell si aprirà la vera shell di Linux.



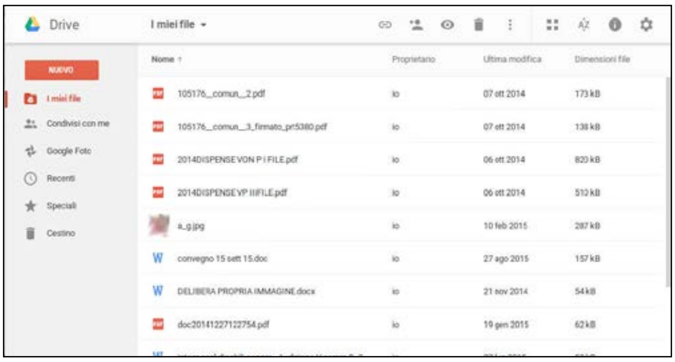
4 Installare Chromium OS

Attenzione! Il prossimo comando cancellerà tutti i dati presenti nel disco interno, come detto prima, non abbiamo avuto molta fortuna col dual boot. Se vi viene chiesta una password utilizzate **password**, poi nella shell lanciate $/usr/sbin/chromeos--install -- dst //dev/sda e avviate l'installazione. Una volta finito togliete la penna USB e riavviate il PC.



5 Ottenere le API keys

Alcune parti del SO non potranno accedere ai servizi Google senza delle chiavi firmate. Effettuate una sottoscrizione al gruppo di Google **http://bit.ly/LXFChromium**, accedete a **https://console.developers.google.com** e scegliete **Create Project**. Abilitate il drive API/SDK, Google Maps Geolocation API e Google Now per Chrome API.



6 Aggiungere le chiavi

Le chiavi sono case sensitive, quindi inseritele facendo attenzione a maiuscole e minuscole. Per farlo aprite **Crosh**, $ shell, $ sudo su, $ mount -o remount,rw/, $ cd/etc, $ vi chrome_dev.conf in **vi** inserite le chiavi come spiegato nel box. Premete **Esc** wq!, fate logout e rifate login, ora **Files** di Chromium dovrebbe funzionare correttamente. LXP

PulseAudio: servire l'audio

Ecco alcuni trucchetti meno conosciuti per **PulseAudio** che permetteranno al suono di propagarsi nella vostra rete per uno streaming remoto

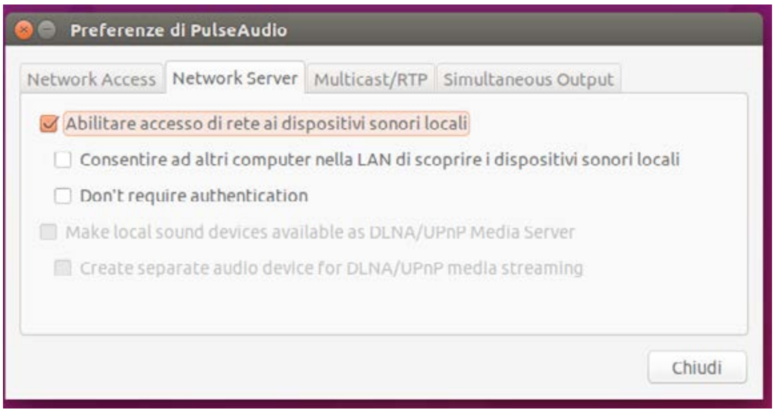
Quando PulseAudio è atterrato nei vostri sistemi, intorno al 2008, c'è stato ogni genere di lamentela da parte degli utenti per il suono non corretto o addirittura sistemi instabili. "Ridateci ALSA" era il grido di chi non sapeva che, in effetti, ALSA non era andato da nessuna parte e che è ancora il metodo preferito dal kernel per comunicare con hardware audio. PulseAudio è un demone, solitamente lanciato come utente, che gestisce l'accesso (ok, monopolizza) ai dispositivi ALSA. Se avete sempre e solo un'applicazione alla volta che utilizza l'audio e avete un solo output audio, allora potete fare tranquillamente a meno di PulseAudio. Tuttavia è una situazione molto improbabile al giorno d'oggi. Non solo perché certi siti vi vogliono bombardare con annunci altisonanti e in autoplay mentre state cercando di ascoltare Spotify, ma anche perché molti utenti vogliono beneficiare dell'audio digitale tramite HDMI contemporaneamente al classico audio analogico tramite jack stereo. Ogni output potrebbe avere il proprio livello: HDMI potrebbe uscire tramite il vostro televisore o monitor, e l'output analogico con il vostro Hi-Fi degli anni '90. Seppure ALSA sia in grado di gestire perfettamente molteplici dispositivi audio, PulseAudio può fare molto di più. Per esempio, supponete di voler unificare i vostri output digitale e analogico (ovvero farli suonare contemporaneamente). Con PulseAudio è semplice. Il mixer in GTK **pavucontrol** vi dà una discreta quantità di controllo su Pulse, quindi usatelo in primis per assicurarvi che il dispositivo audio interno sia impostato per l'output su HDMI. PulseAudio fa riferimento a cose che generano segnali audio come sorgenti, e cose che convertono tali segnali in rumori come sink. Potete elencare i sink disponibili usando il comando

```
$ pacmd list-sinks
```

Questo vi mostrerà tutte le possibili informazioni sui vostri dispositivi, ma ciò che vi interesserà di più saranno i numeri device e subdevice per gli output digitale e analogico. Nel caso d'esempio il dispositivo analogico era il numero 0 con subdevice 0, riferito quindi come **hw:0.0** sotto. Il dispositivo digitale è invece **hw:1.3**. Definite un nuovo sink chiamato **combined** e rendetelo default aggiungendo le righe seguenti a ***/etc/pulse/default.pa***, uno dei file di configurazione principali di PulseAudio:

```
load-module module-alsa-sink device=hw:0,0 sink_name=analogue
load-module module-alsa-sink device=hw:1,3 sink_name=digital
load-module module-combine-sink sink_name=combined slaves=analogue,digital
set-default-sink combined
```

Ora tornando a **pavucontrol** vedrete un nuovo output chiamato *Simultaneous output to Internal Audio Digital Stereo (HDMI), Internal Audio* o qualcosa di simile. Ora, nel riprodurre l'audio, potete andare nella scheda **Riproduzione** e scegliere tale output combinato per qualsiasi applicazione vogliate. Se avete una versione sufficientemente recente di PulseAudio potete fare tutto questo con il programma **paprefs**, vi basterà spuntare la checkbox **Simultaneous Output**. Se non siete dei fan delle interfacce grafiche come paprefs e pavucontrol, ne esistono le versioni console: controllate **ponymix** e **pamixer**. Date un'occhiata anche alla man page di **pacmd**, utilizzabile per lanciare comandi dalla shell.



› Lo strumento *paprefs* vi dà accesso alle caratteristiche di rete di *PulseAudio* senza che dobbiate sporcarvi le mani con la riga di comando

Audio in rete

Una delle caratteristiche principali di PulseAudio è la sua capacità di riprodurre flussi audio da macchine remote. Lo streaming di audio non compresso richiede una connessione affidabile, audio PCM grezzo di qualità CD (stereo, 16 bit, 44.100 KHz) richiede una banda passante di circa 1,4 MB/s. Quindi se avete una wireless non troppo solida potreste preferire il setup di un server MPD o qualcosa di simile, con della musica archiviata in locale controllata da remoto: in questo modo non affiderete ogni frase e movimento della Quinta di Beethoven all'etere, dove vivono i gremlin rubapacchetti. Dal momento che PulseAudio è un esempio di classica architettura client/server, questa funzionalità è relativamente semplice. Tanto per esser chiari, si farà riferimento alla macchina che riproduce l'audio come server

Fare streaming da e verso altri dispositivi media

Paprefs permette anche ai client di fornire l'audio a dispositivi Apple AirTunes (dalla scheda **Network Access**). Se questa opzione è in grigio (e volete veramente fare streaming alla vostra Apple TV o Kodi), potrebbe servirvi il pacchetto **pulseaudio-module-raop**. Paprefs vi permette anche di impostare la vostra macchina come server DLNA/UPnP così da riprodurre l'audio in diversi dispositivi home. Le opzioni rilevanti saranno disattivate finché non installerete il pacchetto **rygel** che fornisce le funzionalità UPnP. Installato questo potete abilitare entrambe le opzioni, la seconda vi permette di avere un controllo separato per il volume dello stream. Rygel avrà bisogno di alcune configurazioni per essere utilizzabile: dovrete abilitare i plug-in **External** o **GstLaunch** in **/etc/rygel.conf** e, a seconda della vostra distribuzione, lanciare **rygel** manualmente. Rygel può comprimere l'audio in diversi formati, il che è cosa buona per l'impegno della vostra banda. Anche se usate il formato lossless FLAC consumerete metà dei bit per secondo. Configurare Rygel per usare il plug-in GstLaunch richiede una sezione di codice simile a questo nel vostro **/etc/rygel.conf**:

```
[GstLaunch]
enabled=true
launch-items=mypulseaudiosink
mypulseaudiosink-title=Audio on @
HOSTNAME@
mypulseaudiosink-mime=audio/flac
mypulseaudiosink-launch=pulsesrc
device=upnp.monitor ! flacenc
```

Ciò abiliterà la riproduzione dell'audio dal vostro server PulseAudio in Kodi o RaspBMC.

e quella che lo invia come client. È possibile fare stream anche da macchine non Linux, e potete usare **zeroconf** in modo che il vostro server si identifichi come individuabile, senza il tedio di dover scrivere indirizzi IP. Tutto il processo può essere infatti completato senza modificare alcun file di configurazione, ma visto che vi piace scrivere e alcuni potrebbero volerlo impostare in un server headless, verrà coperto anche il primo metodo.

Per cominciare, PulseAudio richiederà l'abilitazione del proprio modulo TCP. È semplicemente questione di aggiungere la seguente riga al file **/etc/pulse/default.pa**:

```
load-module module-native-protocol-tcp
```

Come misura di sicurezza, PulseAudio per default non accetterà connessioni da macchine remote a meno che non possiedano lo stesso cookie del server. Potete trovare questo piccolo file in **~/.config/pulse/cookie** quindi copiatelo dal server al client, o viceversa. Se avete accesso via SSH al server (assumendo per esempio che abbia l'indirizzo **10.0.1.1**) potete farlo dal client sostituendo **user** con lo username del server):

```
$ scp user@10.0.1.1:/home/user/.config/pulse/cookie ~/.
config/pulse/cookie
```

Se non vi preoccupa la sicurezza o non volete mantener sincronizzati i cookie potete invece inserire l'indirizzo IP del client (per esempio **10.0.1.2**) in whitelist aggiungendo le opzioni seguenti alla riga **load-module** soprastante:

```
auth-ip-acl=127.0.0.1;10.0.1.2 auth-anonymous=1
```

Potete anche specificare sottoreti nella lista di controllo dell'accesso, per esempio usare **10.0.1.0/24** per abilitare tutti gli indirizzi nella forma 10.0.1.x

Usare Avahi per l'individuazione

Per essere individuabile da altri client, dovete abilitare il servizio **avahi** in entrambe le macchine. Come nota a margine, Avahi è un protocollo di individuazione simile a Bonjour di Apple, scritto anche da non meno che Lennart Poettering, il principale sviluppatore di PulseAudio e Systemd. Avahi, diversamente da questi ultimi, sembra aver fallito nel sollevare molte controversie. Dovrete lanciare Avahi sia nel client che nel server. Se state usando Systemd lo potete fare con

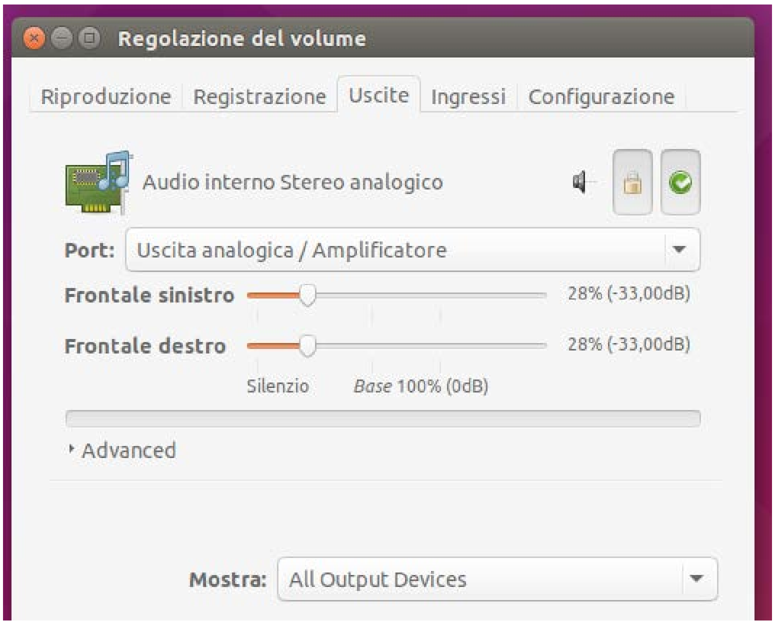
```
$ sudo systemctl start avahi-daemon
```

analogamente per altri sistemi di init:

```
$ sudo service avahi-daemon start
```

Ora dovete configurare PulseAudio per usare Avahi in entrambe le macchine. Sul server dovrete aggiungere la riga seguente a **/etc/pulse/default.pa**:

```
load-module module-zeroconf-publish
```



› Se va tutto bene il vostro server sonoro dovrebbe mostrarsi così in *pavucontrol*. Potete quindi dirigere tutto l'audio o solo alcune applicazioni particolari

e sul client aggiungete questa riga:

```
load-module module-zeroconf-discover
```

Ora, quando lancerete **pavucontrol** nella macchina client, vedrete il server nella scheda **Uscite**. Potete dirigervi tutto l'audio o solo l'output di alcune applicazioni. Si è detto che potete farlo senza modificare alcun file, e non era una bugia. Dovrete installare **paprefs** sia sul client sia sul server. Nel client dovrete spuntare la prima casella nella **Network Access**, il che permetterà l'individuazione dai server PulseAudio nella rete. Sul server andate nella scheda **Network Server** e spuntate la prima (**Abilitare accesso di rete ai dispositivi sonori locali**), la seconda (**Consenti ad altri computer nella LAN di scoprire i dispositivi sonori locali**) e (se non volete mettere le mani nei cookie di PulseAudio) la terza casella. Distribuzioni diverse impacchettano PulseAudio diversamente, ma se trovate difficoltà potete sempre provare a riavviare il servizio **avahi-daemon**. Il server PulseAudio ascolta sulla porta TCP 4713, quindi assicuratevi che non ci siano ostacoli con il firewall. Potete provare forzando la variabile **PULSE_SERVER**. Per esempio, lanciate il media player **Totem** con il vostro server:

```
$ PULSE_SERVER=10.0.1.1 totem
```

Ecco come impostare un flusso audio che gira su un client remoto. LXP

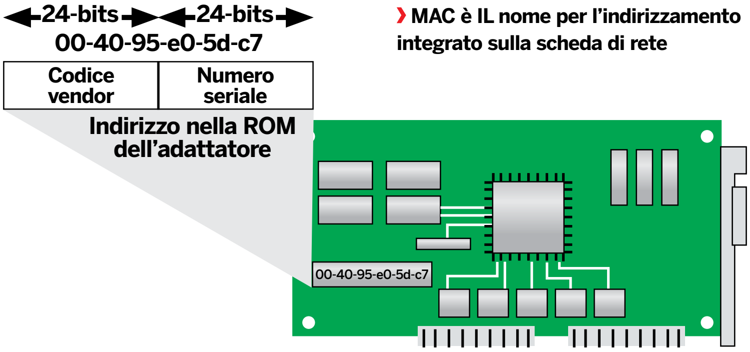
Tcpdump: cattura dei dati

Un'introduzione al protocollo di comunicazione di rete TCP/IP e come catturare i dati usando lo strumento da riga di comando **tcpdump**

Quale miglior modo di imparare come usare tcpdump che usarlo per scoprire come i computer comunicano attraverso la rete. Questo tutorial combina i comandi di tcpdump e la teoria delle reti. Usando questi comandi recupererete dati dalla rete e dimostrerete i componenti chiave usati per la comunicazione TCP/IP. I dati viaggiano tra varie reti, di dispositivo in dispositivo, in unità chiamate *Protocol Data Unit* (PDU) e la composizione di una PDU dipende dal protocollo di rete. Pensate a una singola PDU come a un vagone merci sui binari, che trasporta merci in forma di dati, e ogni PDU è diversa a seconda del tipo di dato che trasporta. Tcpdump è un'interfaccia a riga di comando usata per catturare i contenuti delle PDU in viaggio su una rete. Questo strumento funziona senza impattare sui contenuti della PDU stessa, un po' come guardare dei pesci che nuotano in un acquario. Prima di usare tcpdump dovete acquisire una conoscenza basilare della comunicazione dati. Tale conoscenza vi servirà come punto di riferimento per svilupparne di nuova sulla comunicazione strutturata di dati. Può essere usata per spiegare l'operatività di tcpdump sulle PDU. Un modello per la comunicazione strutturata punto a punto è l'*Open Systems Interconnection* (OSI). Tale modello offre una base comune per il coordinamento degli standard allo scopo di intercomunicazione tra sistemi. Le funzioni del modello OSI sono organizzate in una gerarchia di livelli (layer). Ogni layer esegue un sottoinsieme di funzioni per il totale richiesto dalla comunicazione, si basa su servizi forniti dai layer più bassi e fornisce servizi ai layer più alti. Tale modello è uno strumento utile per affrontare il tema ed esaminare protocolli specifici: guardate l'illustrazione alla pagina seguente per capire tutti i layer di OSI. Il disegno è stato personalizzato per concentrarsi sui quattro layer del protocollo TCP/IP, e tcpdump può analizzare tali informazioni. Nella parte sinistra dell'illustrazione avete un computer che vuole comunicare qualcosa via rete. Un messaggio entra nel sistema via software nel layer **Applicazione** e il protocollo di tale layer incapsula il messaggio con alcune informazioni di controllo, quindi passa la PDU al layer **Presentazione**. Quest'ultimo tratta i dati, incapsula alcune informazioni in un header e passa a sua volta la PDU al layer **Sessione**. Come vi aspetterete, il layer **Sessione** stabilisce una sessione con informazioni di controllo nell'header e gira quindi la PDU al layer **Trasporto**. Il layer **Trasporto** spezza le informazioni dei layer precedenti in segmenti e incapsula le informazioni di controllo con un header, quindi passa la PDU al layer **Rete**. Questo layer incapsula i segmenti dei layer precedenti con un indirizzo di rete utilizzabile per il routing, passando poi il pacchetto al layer **Data Link**. Quest'ultimo incapsula tutte le informazioni di cui sopra con un header e un trailer, applica un indirizzo fisico e informazioni per il controllo d'errore, girando il tutto al layer **Fisico**, che materialmente mette le informazioni del frame sul cavo. Wow! Ora spostatevi a destra dell'illustrazione per seguire la ricostruzione del messaggio fino alla destinazione, esaminando i primi quattro livelli in relazione al protocollo TCP/IP. Questo lato mostra i dettagli su come le PDU sono costruite con i diversi tipi di campo, le dimensioni dei campi in byte e i nomi comunemente usati.

Tip

Alcuni sostengono che usare **tcpdump** per l'analisi di rete è come usare **vi** per l'editing di testo. Tali strumenti mettono oneri e onori nelle mani dell'utente, mentre altri semplificherebbero il lavoro.



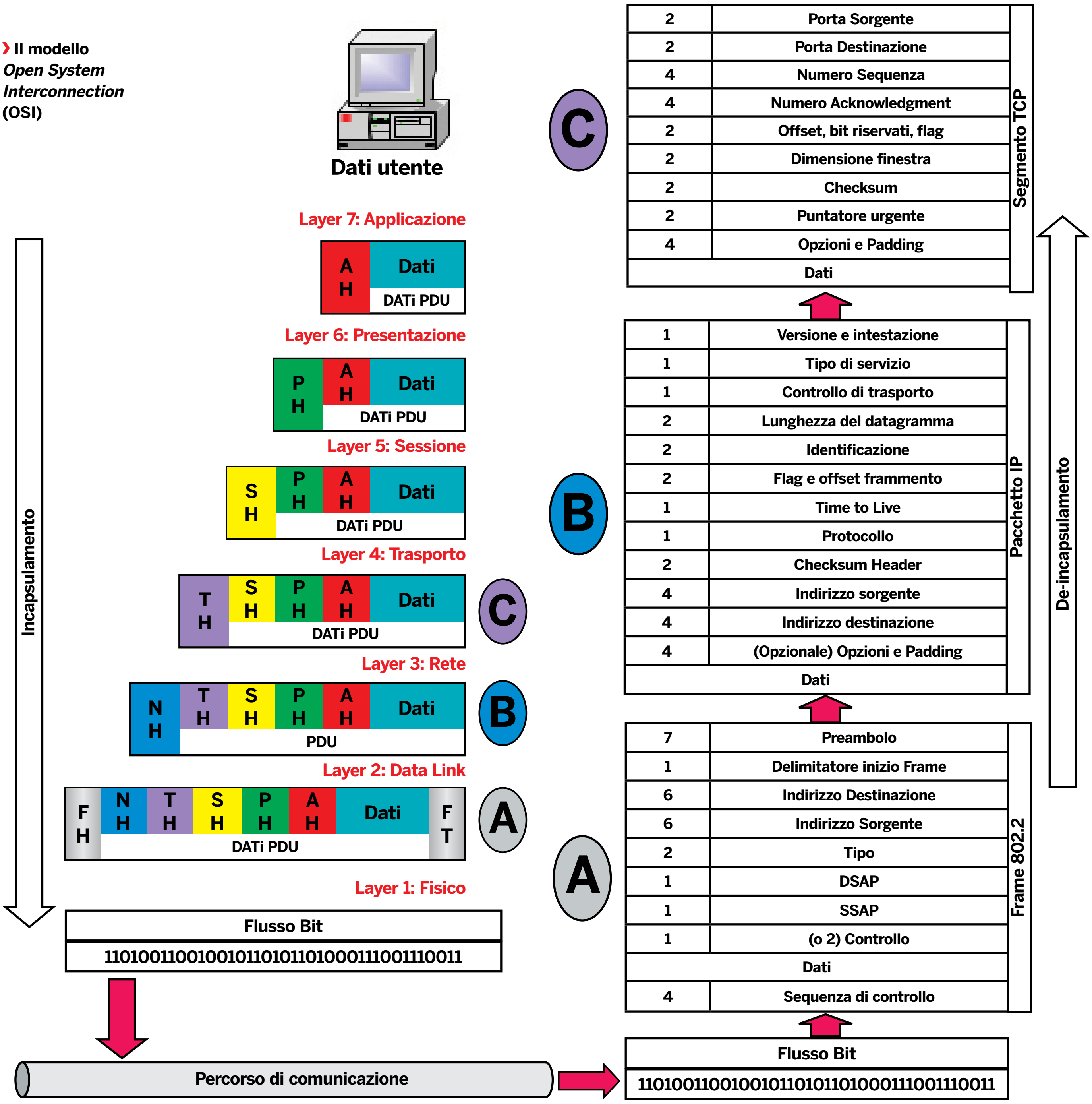
› MAC è IL nome per l'indirizzamento integrato sulla scheda di rete

L'indirizzo *Media Assess Control* (MAC) identifica il produttore (codice vendor) della scheda di rete e l'ID unico del dispositivo (numero seriale)

Ancora sui layer

Il layer **Fisico** è dove viene trasmesso effettivamente il flusso di bit 0 e 1. Questo layer offre la funzionalità e le procedure per spostare cifre binarie sul media fisico. In una rete LAN, le cifre binarie possono essere segnali elettrici o impulsi luminosi, a seconda se state usato del cavo di rame o fibra ottica, e la scheda di rete (NIC) nel computer deve supportare tali funzionalità. Il layer **Data Link** (vedete **A** nell'illustrazione) fornisce un servizio di comunicazione affidabile tra i collegamenti fisici trasmettendo i dati in blocchi PDU chiamati frame. I frame offrono dati temporali, flusso e controllo dell'errore in campi di dimensioni diverse definite

› **Il modello *Open System Interconnection* (OSI)**



dagli standard relativi. Il layer **Rete** (**B** nel diagramma) fornisce servizi di switching e trasmissione tra i due sistemi, usando strutture nelle PDU chiamate pacchetti. Il layer **Trasporto** (**C**) si occupa della comunicazione affidabile punto a punto attraverso correzione dell'errore e controllo di flusso in PDU chiamati segmenti. Il diagramma dimostra il concetto di incapsulamento dei dati. Viene effettuato mostrando i bit nel cavo del frame Ethernet che contiene il pacchetto IP che contiene il segmento TCP. Le PDU del frame, del pacchetto e del segmento seguono una specifica che permette a strumenti quali tcpdump di analizzare posizioni specifiche (ovvero byte). I dati si spostano nella rete come 1 e 0 in impulsi su rame o luce in fibra ottica. Gli uni e gli zeri nel cavo costruiscono un frame, e i frame passano da macchina a macchina usando l'indirizzamento MAC. Indirizzo MAC, fisico, hardware o Ethernet sono nomi diversi per l'unico indirizzo assegnato al dispositivo fisico. Le connessioni individuali sulla scheda di rete hanno ognuna un indirizzo MAC e un frame, come già detto, e lo sfruttano per saltare di sistema in sistema. Quando un frame deve spostarsi su un'altra rete (per esempio attraverso un router), l'indirizzo MAC del frame viene modificato per riflettere il punto di salto. Host_X vuole comunicare con Host_Y. Gli indirizzi IP, se mostrati, indicano che gli host sono su reti diverse. Host_X invia un frame con l'indirizzo MAC di destinazione a Router_A. Quest'ultimo invia un frame con l'indirizzo MAC destinazione del prossimo salto a Router_B. Router_B invia un frame con l'indirizzo MAC di Host_Y. Sia Router_A sia Router_B »

» dipendono dall'indirizzo IP nel pacchetto incapsulato per decidere su che percorso indirizzare. Il frame è costruito con indirizzi MAC diversi per sorgente e destinazione, ma il payload del pacchetto rimane lo stesso durante tutto il viaggio. Gli indirizzi MAC vengono talvolta riferiti come 'next hop address', dal momento che l'indirizzo porta il frame solamente al prossimo salto.
Potete visualizzare gli indirizzi MAC del vostro sistema Linux lanciando ifconfig da riga di comando:

```
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 04:1e:64:f0:1c:63
        inet6 fe80::61e:64ff:fef0:1c63%en1 prefixlen 64 scopeid 0x4
        inet 192.168.1.134 netmask 0xffffff00 broadcast 192.168.1.255
...
```

L'indirizzo MAC è il testo che segue **ether**. Per mostrare l'indirizzo MAC del router usato dalla rete scrivete:

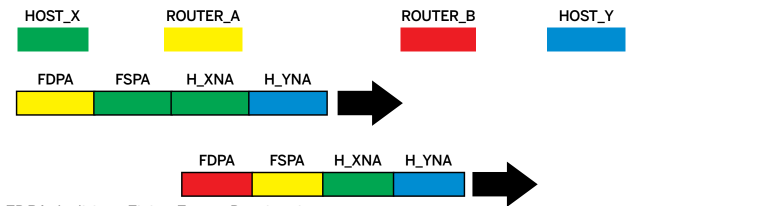
```
$ arp -a
? (192.168.1.129) at d8:30:62:51:7:b2 on en1 ifscope [ethernet]
```

È importante leggere l'avviso nel box nella pagina accanto (**Policy aziendale**) prima di usare tcpdump. Per generare i dati per questo tutorial, il sistema sorgente è un desktop che usa un browser mentre la destinazione è un server su cui gira un web server. Entrambi gli IP sorgente e destinazione sono sulla stessa rete (per risparmiare spazio l'output di diversi comandi è stato troncato per offrire testo specifico).
Se volete seguire questo tutorial dovrete sostituire il nome della scheda di rete (en1) dal computer di test con quello che trovate nell'output del vostro ifconfig. Per avere una spiegazione completa degli switch andate su **www.tcpdump.org**. Usando tcpdump, gli indirizzi MAC sia dell'host (sorgente) sia del router (destinazione) vengono visualizzati usando un frame di dati. Riferendosi al diagramma OSI, l'indirizzamento MAC per la sorgente e la destinazione è quello che appare nel frame del layer Data Link:

```
$ sudo tcpdump -e -i en1
11:31:50.423825 00:60:3b:34:ab:32 (oui Unknown) > 04:1e:64:f0:1c:63 (oui Unknown), ethertype IPv4 (0x0800), length 66: edge-star-shv-01-mxp1.facebook.com.https > 192.168.1.134.50250: Flags [.], ack 2713988966, win 2043, options [nop,nop,TS val 1769913262 ecr 206834912], length 0
```

I segni > possono essere usati per determinare la sorgente e la destinazione per la PDU. Non vi allarmate se l'output scrolla nello schermo. Ricordate che l'interfaccia sta operando in modo promiscuo quindi visualizza qualsiasi cosa sniffi. Per uscire da tcpdump premete **Ctrl+C**.

› **Un frame con solo un indirizzo fisico che incapsula un pacchetto con l'indirizzamento IP**



FDPA: Indirizzo Fisico Frame Destinazione
FSPA: Indirizzo Fisico Frame Sorgente
H_XNA: Indirizzo di rete Host_X (datagram)
H_XNA: Indirizzo di rete Host_Y (datagram)
Nota: L'indirizzo fisico cambia quando il datagram viaggia di salto in salto, mentre l'indirizzo di rete rimane il medesimo

Guardare le PDU

Usando tcpdump potete vedere gli indirizzi IP sia dell'host sia del Web server. Notate dall'illustrazione OSI gli indirizzi IP che sono l'indirizzamento per sorgente e destinazione che appare nel pacchetto.

```
$ sudo tcpdump -n -i en1
17:34:36.468664 IP 192.168.2.253.38412 > 192.168.2.225.http: Flags [S], seq 2642752770, win 29200, options [mss 1460,sackOK,TS val 332382530 ecr 0,nop,wscale 7], length 0
```

Se aggiungete una seconda n al comando (-nn), oltre a fare il DNS lookup per quell'indirizzo IP, verrà anche fatta la risoluzione dei numeri di porta in numero. Il nome del servizio HTTP sarà tradotto in 80. Rimuovete lo switch -n da tcpdump per visualizzare gli hostname dell'host e del sito Web:

```
$ sudo tcpdump -i enp3s0
# 17:30:27.428424 IP mybox.38409 > web server.http: Flags [S], seq 1847458607, win 29200, options [mss 1460,sackOK,TS val 332133490 ecr 0,nop,wscale 7], length 0
```

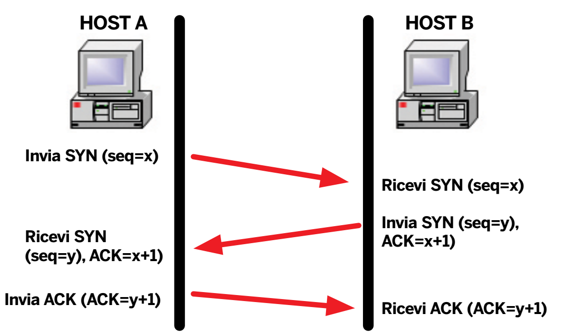
Avete problemi a trovare quella PDU con quello specifico indirizzo IP? Aggiungete lo switch per l'host al comando e verrà visualizzato solamente il traffico che contiene quello specifico indirizzo IP. Ecco le parti chiave di una PDU mostrate in HEX e ASCII per uno specifico IP usando tcpdump:

```
$ sudo tcpdump -X -i en1 host 192.168.2.225
# listening on en1, link-type EN10MB (Ethernet), capture size 65535 bytes
17:53:24.208005 IP mybox.38422 > web server.http: Flags [S], seq 458400957, win 29200, options [mss 1460,sackOK,TS val 333510270 ecr 0,nop,wscale 7], length 0
0x0000: 4500 003c 52c3 4000 4006 60ca c0a8 02fd E..<R.@.@.`.....
```

Al layer **Rete**, l'indirizzamento IP è un campo chiave da esaminare nel pacchetto. Potete far sì che tcpdump controlli degli specifici IP usando switch che descrivano

Inconsistentemente consistente

Quando un termine non è usato consistentemente, o vengono usati diversi termini per indicare la stessa cosa, esiste una buona possibilità per un novello amministratore di sistema di essere frustrato. Un esempio è il riferimento ai pacchetti che viaggiano in rete. Questo termine è più o meno accurato: in effetti, i frame viaggiano sulla rete e trasportano il payload del pacchetto. Nell'illustrazione di un frame (*sopra*), l'indirizzamento è indicato come sorgente e destinazione. Tale indirizzo può essere chiamato indirizzo *Media Access Control* (MAC), indirizzo fisico, indirizzo hardware o indirizzo Ethernet, e tutti questi termini sono appropriati a seconda del punto d'origine dell'informazione.
I sysadmin dovrebbero cercare di usare il termine corretto, non quello più cool. Gli standard come il modello OSI sono un buon posto per iniziare a cercare un riferimento, aiutando ad aumentare la chiarezza. Usando la vostra conoscenza assieme a tcpdump, potete esplorare alcune delle funzioni operative del protocollo TCP/IP. Le funzioni hanno una struttura di campi in una posizione definita che tcpdump individua e mostra. Le funzioni operative permettono la consegna di servizi, come creare una connessione a un sito Web o l'invio di dati tra i punti.



› Seguite la freccia tra le tre PDU per scoprire i componenti del *three-way handshake*

l'accesso, per esempio sotto è stato abilitato tcpdump per controllare degli specifici campi nell'indirizzo, sorgente e destinazione o controllare tutto il traffico in una sottorete:

```
$ sudo tcpdump -i en1 src 192.168.2.253
$ sudo tcpdump -i en1 dst 192.168.2.253
$ sudo tcpdump -i en1 net 192.168.2.0/24
```

Passate lo stack di protocollo del layer che contiene gli indirizzi IP al layer **Trasporto**. A questo livello viene eseguita la segmentazione con le porte sorgente e destinazione. I campi porta sono punti di accesso individuali a servizi specifici. La segmentazione divide i dati in segmenti, fornendoli di numeri di controllo sequenza e acknowledgment per permettere la ricostruzione dei dati.

Individuazione della porta

Due esempi di porta sono 80 per HTTP e 22 per SSH. Un browser Web può contattare un server Web sulla porta 80, ma se il server è sicuro la porta sarà la 443. Tali porte sono standard. Alcuni siti Web si offuscano offrendo il servizio (per esempio, SSH) su porte diverse, così un attaccante deve scansionare tutte le possibilità per scovare il servizio. Il file **/etc/services** elenca i servizi e le porte conosciute associate, e tale file è usato per indicare i nomi dei servizi quando viene usato in tcpdump lo switch per la risoluzione dei nomi `-nn`). Le porte sono i punti dove vengono stabilite le connessioni ai servizi, e usando `netstat` potete vederne l'attività:

```
$ netstat -an | more
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign
Address       State
tcp       0      0 0.0.0.0:22          0.0.0.0:*          LISTEN
tcp       0      0 192.168.2.225:22
192.168.2.253:53643  ESTABILISHED
```

Nella prima riga il servizio SSH ascolta per richieste di connessione e nell'altra una connessione SSH è in corso all'indirizzo IP del server Web. Potete filtrare in tcpdump usando lo switch di riferimento della porta:

```
$ sudo tcpdump -X -i en1 port 80
```

Potete includere anche i descrittori `scr` e `dst` come switch alle porte per limitare ulteriormente i risultati:

```
$ sudo tcpdump -X -i en1 src port 8080
$ sudo tcpdump -X -i en1 dst port 50001
```

TCP/IP è un protocollo affidabile, e le funzioni di protocollo che lo rendono tale risiedono nel layer **Trasporto** del modello OSI nella suite *Transport Control Protocol* (TCP). In aggiunta al supporto delle porte per la connessione a servizi conosciuti, esistono delle flag per le connessioni stabilite e numeri di sequenza e acknoledgment per gestire i segmenti (vedete il diagramma relativo).
Per espandere la vostra conoscenza vedrete ora come un host stabilisce una connessione a un server Web usando TCP/IP. Prima di inviare dati è necessario stabilire un contatto tra i due host. I dati inviati devono essere spezzati in PDU per il trasporto. A livello Trasporto un campo SYN in un segmento apre una richiesta di connessione e un campo FIN la chiude. I segmenti inviati sono ordinati da numeri di sequenza e usano gli acknoledgment per notificare l'altro capo che tutti i segmenti richiesti sono stati correttamente ricevuti. Se mancano dei segmenti durante il trasporto è possibile richiederne il re-invio. TCP/IP stabilisce una connessione affidabile usando il cosiddetto *three-way handshake*. I campi flag, numero di sequenza e acknowledgment nel diagramma (in alto a sinistra) sono posizionati con dati specifici catturati usando tcpdump, e la comunicazione inizia con l'invio da parte di Host A di un segmento contenente una flag SYN.
Un modo per impiegare tcpdump nell'analisi di un problema è catturare i dati in un file per un successivo esame. Per esempio, potete creare un file binario da tcpdump (**tcpdump_output**) usando:

```
$ sudo tcpdump -nnvvvXS -i enp3s0 src 192.168.2.253 or
src 192.168.2.225 -w /tmp/tcpdump_output
```

Per leggere i contenuti del file usando tcpdump usate:

```
$ sudo tcpdump -nnvvvXS -r /tmp/tcpdump_output
```

I dati prodotti da tcpdump usano un formato binario supportato da altri strumenti come **Wireshark** e **Snort**. Durante una lezione in un istituto tecnico, il docente ha chiesto agli studenti di spiegare cosa si aspettavano da una misura, prima di usare l'attrezzatura di test. Per farlo gli studenti dovevano capire cosa stavano cercando, invece che sperare di trovare 'qualcosa'. Il suggerimento del docente fu, ed è, un buon modo per incoraggiare gli studenti a usare la loro conoscenza prima della pratica. Se dopo aver misurato, questo coincide con le vostre aspettative, potete andare oltre e provare un diverso punto di vista. Queste capacità richiedono tempo per essere padroneggiate. L'analisi del traffico di rete è complicata e l'utilizzo efficace di tcpdump richiede una comprensione solida dei costrutti del trasferimento di dati. Quale modo migliore di ottenere tale conoscenza che usare gli strumenti che vi richiedono di definire cosa state cercando prima di cercarlo effettivamente? Tcpdump è la pietra d'angolo per imparare e fortificare le vostre conoscenze sulla comunicazione di rete. LXP

Tip

Catturare dati destinati a un sito Web la cui URL inizia con HTTP può mostrarvi username e password nella sezione di payload della PDU.

Policy aziendale

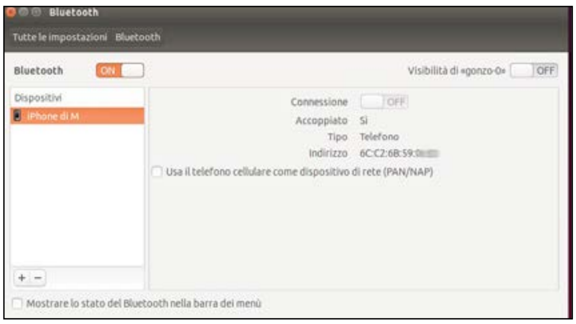
Per permettere a tcpdump di catturare dati sulla rete, la scheda deve operare in modalità promiscua, il che significa che invece di accettare i pacchetti destinati solamente al proprio sistema, diventa indiscriminata e mostra tutti i dati catturati. Molto traffico della rete non è cifrato. Usare questo strumento può darvi accesso a dati non richiesti; dati quali username e password che passano in rete. Per questa ragione usare tcpdump in ambiente lavorativo potrebbe violare le policy aziendali. Controllate la policy di sicurezza della vostra azienda prima di usare un programma sulla rete di lavoro. Senza autorizzazione l'azienda potrebbe avere delle basi per il licenziamento.

I consigli per la sicurezza

Se non potete tenere i vostri portatili o cellulari lontano dalle mani dei ladri, lo staff di Linux Pro vi mostrerà come mettere al sicuro i dati al loro interno



Spesso parliamo di sicurezza pensando alla gestione delle password o alla crittografia dei dati, ma i dispositivi portatili presentano un altro potenziale problema, il furto fisico. È facile dire che bisogna stare attenti a non farsi rubare i propri dispositivi, ma cosa si può fare per limitare i danni se il vostro portatile o cellulare viene rubato? Vedremo alcuni modi per mantenere i dati al sicuro nel caso in cui cadessero in mani sbagliate, evitando di aggiungere tanti livelli di sicurezza da renderlo inutilizzabile anche per voi.



› **Prima di poter utilizzare un device con *BlueProximity* è necessario accoppiarlo con il proprio computer utilizzando l'interfaccia offerta dal proprio ambiente desktop**

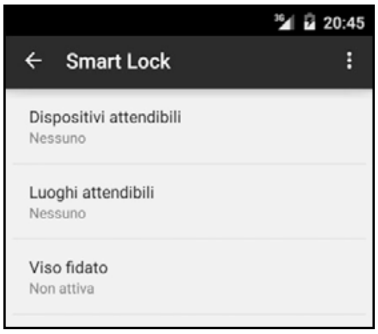
Le password robuste sono sempre una buona cosa, soprattutto nel caso di device mobili. Se qualcuno prende il vostro portatile, sarà libero di provare a trovare la vostra password quante volte vuole. Alcuni programmi come **John the Ripper** possono tentare attacchi brute-force sulle vostre password garantendo al malintenzionato di turno accesso ai vostri dati. Non potete fare molto riguardo a questo ma potete rendere illeggibili i vostri file più importanti. Richiedere l'accesso come root per accederci non è sufficiente perché basterà avviare il PC con un CD di recupero per poter leggere tutti i file come utente root, quindi crittografare il disco dovrebbe essere una priorità. Abbiamo già visto in un articolo passato come applicare la crittografia alla propria directory **home**, ma applicare crittografia a tutto il disco significa anche dover reinstallare o fare un backup di tutto e crittografare il disco con **dm-crypt**. È un sacco di lavoro, ma dovreste valutare l'ipotesi quando fate una nuova installazione della vostra distribuzione. È bene anche inserire una password al BIOS per evitare che qualcuno cambi le impostazioni e lo avvii da un altro device.

Bloccare tutto

La crittografia di un disco o della cartella home è solo una parte della soluzione. Ferma un malintenzionato che vuole utilizzare un CD live per leggere i vostri dati, ma se il portatile viene rubato mentre è acceso, tutti i vostri dati saranno disponibili. Esiste una soluzione semplice a questo inconveniente? Beh, dato che il modo più comodo per trasportare un portatile è con lo schermo chiuso, possiamo fare in modo che il computer si blocchi quando chiudiamo lo schermo? La risposta è sì: il sistema ACPI reagisce a diversi eventi, inclusa la chiusura e l'apertura dello schermo. Spesso viene utilizzato per spegnere lo schermo e aumentare la durata della batteria, ma potete anche utilizzarlo per bloccare il computer. I dettagli cambiano da desktop a desktop; in KDE potete impostare questa azione nella sezione **Power Management** delle **Impostazioni**, invece gli utenti Gnome dovrebbero guardare in **Privacy**, per gli utenti Unity invece sotto **Sicurezza e privacy**. Potete impostare questa opzione anche da soli modificando la variabile di ambiente. Gli utenti **Systemd** possono modificare il file **/etc/systemd/logind.conf** e impostare `HandleLidSwitch` a 'lock'. Alternativamente potete utilizzare direttamente ACPI per gestire la cosa, installando **xlock** dal vostro gestore di pacchetti e creando un file **/etc/acpi/events/lid** contenente

Le opzioni di Android

Abbiamo considerato principalmente portatili con Linux in esecuzione, ma i dispositivi mobili con Android possono essere persi con facilità. Proteggere il dispositivo con un PIN o un pattern per sbloccarlo è sicuro ma non molto comodo. Fortunatamente, l'edizione **Lollipop** di Android (5.x) offre una funzionalità chiamata **Smart Lock**, che potete trovare negli **Agenti di attendibilità** del menu **Sicurezza** delle impostazioni, e permette di bypassare il blocco quando sussistono certe condizioni. Per esempio in base alla posizione GPS o alla presenza di un device Bluetooth. Per esempio, potete disabilitare il blocco quando siete a casa o nella vostra auto. Attraverso la **Gestione dispositivi Android** (all'indirizzo **https://www.google.com/android/devicemanager?hl=it**) si può localizzare o addirittura ripulire un telefono perso. Potete rendere il vostro telefono più sicuro abilitando due fattori di autenticazione e impostando la richiesta del PIN all'accensione.



› **Lo smart lock di Android Lollipop ci permette di sbloccare il dispositivo in certe particolari circostanze**

```
event=button[ /]lid.*close
action=sudo -u vostroutente /usr/vin/xlock -display :0
```

Questo comando esegue **xlock** per bloccare il display quando si chiude lo schermo, e servirà la vostra password per sbloccarlo. Utilizziamo sudo per eseguirlo come il vostro utente perché il demone ACPI viene eseguito come root e **xlock** vuole la password dell'utente che lo ha lanciato. Dovrete avviare il servizio **acpi** dopo aver creato o modificato questo file. Non tutte le distribuzioni disabilitano la possibilità di usare **Ctrl**+**Alt**+**F1** per accedere a una console virtuale e **xlock** blocca soltanto il display X. Potete risolvere questo problema aggiungendo quanto segue al file **/etx/X11/xorg.conf** o al file **/etc/X11/xorg.conf.d**:

```
Section "ServerFlags"
        Option "DontVTSwitch" "true"
EndSection
```

Prossimità

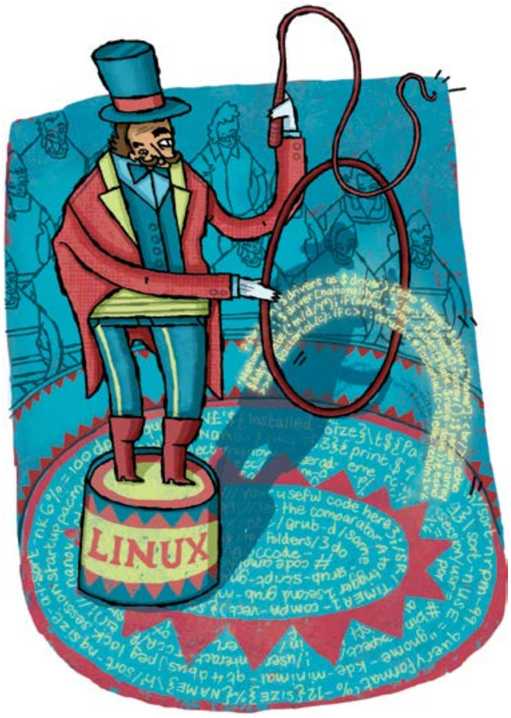
Questo metodo ha due inconvenienti: si affida sul fatto che un ladro chiuderà lo schermo, che non è un grosso problema, e richiede che inseriate la password ogni volta che lo riaprite, e abbiamo detto che è necessario usarne una robusta. Sarebbe bello se il portatile funzionasse quando voi lo state usando ma non quando ci prova qualcun altro. Non stiamo parlando di scanner di impronte digitali ma di qualcosa che praticamente tutti hanno: uno smartphone. **BlueProximity** (**http://blueproximity.sourceforge.net/**) controlla un device Bluetooth accoppiato e quando questo si allontana a una certa distanza blocca il computer. Se il vostro telefono è sempre nella vostra tasca, o in mano, funziona alla grande.
È ancora meglio se lo collegate a uno smartwatch. Il primo passo è quello di accoppiare il vostro device Bluetooth con il computer, utilizzando il tool di impostazioni del vostro desktop. Ora potete aprire **BlueProximity**. Se avviandolo dal launcher non si apre nulla, controllate il system tray perché potrebbe essere partito in background. Iniziate col premere il pulsante **Ricerca dispositivi**. Selezionate il vostro device nella lista di quelli trovati e premete **Utilizza il dispositivo selezionato** che copierà l'indirizzo MAC del device nel campo di testo sottostante. Ora potete impostare i parametri nel tab **Dettagli di prossimità**. Le impostazioni sulla distanza sono in realtà basati sulla potenza del segnale, più alto è il numero più basso è il segnale o maggiore è la distanza. La durata indica per quanto tempo il vostro device deve aver superato la distanza prima che BlueProximity agisca. Se volete impostare una distanza specifica, premete il pulsante **Reset Min/max**, portate il vostro device alla distanza che volete, aspettate qualche secondo e tornate indietro così vedrete in basso sullo schermo la distanza massima registrata. Per molti utenti il default dovrebbe andar bene, normalmente si imposta la distanza di blocco maggiore rispetto a quella di sblocco, altrimenti potreste trovarvi con il PC che si blocca e sblocca continuamente quando vi trovate alla giusta distanza. Anche impostare una durata troppo bassa può generare eventi inaspettati, perché la forza del segnale Bluetooth può fluttuare molto. Il terzo tab permette di impostare quali comandi eseguire quando si blocca e sblocca: di default esegue lo screensaver di Gnome ma potete usare la freccia per utilizzare xscreensaver. BlueProximity non esegue il comando di blocco finché la sua finestra è aperta (anche se a volte a noi è capitato che partisse comunque il blocco), ma modifica il colore della chiave nell'icona per mostrare se si è nel range o meno. Verde e rosso hanno un significato ovvio, giallo significa che si è all'interno del range di blocco, ma all'esterno del range di sblocco. Il comando di prossimità viene eseguito ogni volta che il vostro device entra nel range, per l'intervallo specificato. L'impostazione di default fa sì che parta lo screensaver di Gnome e cancelli lo schermo, anche se non toccate nulla su mouse o tastiera. Potrebbe essere necessario qualche aggiustamento alle impostazioni, ma col tempo otterrete un PC che risponderà come desiderate all'avvicinarsi o allontanarsi del vostro device Bluetooth.

Trovare un device perso

Se dovesse succedere il peggio e il vostro device viene rubato, c'è qualcosa che potete fare per proteggere i vostri dati. Installando il client **Prey** (**http://preyproject.com**) sul vostro computer e sul vostro device mobile, e impostando un account gratuito, verrete avvisati quando il vostro PC va online, anche se non siete voi a farlo. Il servizio base è gratuito, con diversi livelli di servizi premium per funzionalità e device aggiuntivi, l'opzione gratuita funziona con massimo tre dispositivi e offre un buon livello di tranquillità. Esistono altri servizi come **Cerberus**, ma Prey ha il vantaggio di essere Open Source e di funzionare con tutti i desktop e tutti i sistemi operativi mobili. LXP

Le funzioni temporali

Nella quarta puntata della guida alla programmazione di sistema vedrete come affrontare una breve storia del tempo



Immaginerete che il tempo sia un argomento piuttosto semplice. Ammesso che non vogliate tornare al Big Bang o viviate vicino a un buco nero, o viaggiate vicino alla velocità della luce, il tempo scorre, secondo dopo secondo, no? Sì, ma potrebbe essere più complicato di quanto pensiate. Per iniziare dall'inizio... tradizionalmente il tempo nei sistemi Unix e Linux è misurato in secondi, relativi a un momento chiamato "epoch" che è l'esatto inizio dell'anno 1970 (la mezzanotte del 1° gennaio) nella timezone UTC+0. Viene memorizzato come intero, definito usando il typedef time_t. In sistemi a 32 bit un time_t è un intero signed a 32 bit, in sistemi a 64 bit è a 64 bit.
La chiamata di sistema time() ritorna il tempo Unix così come appena definito. Potete quindi stamparlo con:

```
#include <stdio.h>
#include <time.h>
main()
{
  printf("%ld\n", time(NULL));
}
```

Alternativamente, il comando **date** può farlo semplicemente dalla riga di comando così:

```
$ date +%s
1441227689
```

Ovunque siate nel mondo, in qualsiasi timezone, e qualsiasi lingua parliate, se lanciate questo programma nel medesimo istante otterrete la stessa risposta di chiunque altro, fintantoché gli orologi dei computer siano corretti. Non è però una rappresentazione memorabile se state cercando di ricordare che giorno della settimana è o se state cercando di capire se siete in tempo per la metro delle 11:53 per la Garbatella. Questa rappresentazione è però importante, non solo perché è la rappresentazione dei timestamp come registrata negli inode dei file. Se volete una risoluzione maggiore di un secondo, la chiamata gettimeofday() ritorna una struttura timeval che mantiene il tempo in secondi e microsecondi:

```
struct timeval {
        time_t tv_sec; /* secondi */
        suseconds_t tv_usec; /* microsecondi */
};
```

Potete quindi scrivere una versione 'più accurata' del programma:

```
#include <stdio.h> #include <sys/time.h> main()
{
        struct timeval tv;
        gettimeofday(&tv, NULL);
        printf("%ld.%06ld\n", tv.tv_sec, tv.tv_usec);
}
```

Che genererà un output simile a questo:

```
1441227813.892352
```

Tuttavia, non dovreste confondere la risoluzione con l'accuratezza. Al meglio, il vostro computer sarà accurato a qualche decina di millisecondi. Naturalmente, se state usando gettimeofday() per misurare l'intervallo di tempo immediatamente prima di qualche operazione e quindi subito dopo, sottraendo i risultati, l'accuratezza assoluta dell'orologio non importa.

Il tempo come stringa leggibile

Ci sono molti modi per convertire un time_t in una rappresentazione più utile (leggibile). Uno dei più semplici è ctime(). Estendete il vostro programma originale:

```
#include <stdio.h>
#include <time.h>
main() {
```

Tip

Dovreste far girare sempre il demone per il Network Time Protocol con un minimo di tre time server. È facile da impostare e manterrà l'orologio del vostro computer aggiornato con uno scarto di qualche decina di millisecondi.

```
        time_t t;
        t = time(NULL);
        printf("%s\n", ctime(&t));
}
```

Che stampa una riga simile a questa:

```
Wed Sep  2 23:23:30 2015
```

Questa conversione prende in considerazione la vostra timezone ma non usa le impostazioni del locale, per cui i nomi abbreviati di giorno e mese saranno sempre in inglese. Mentre ctime() è un modo economico e veloce di ottenere un tempo leggibile, non ritorna il tempo in modo da farci dell'aritmetica sopra. Per esempio, supponete di voler determinare se un time_t cade prima o dopo mezzogiorno. Per questo, le funzioni gmtime() e localtime() sono più utili. Entrambe ritornano quello che viene definito tempo "destrutturato" (*broken down*) in una struttura tm:

```
struct tm {
        int tm_sec;
        int tm_min;
        int tm_hour;
        int tm_mday;
        int tm_mon;
        int tm_year;
        int tm_wday;
        int tm_yday;
        int tm_isdst;
};
```

I campi sono ampiamente autoesplicativi, anche se non sono tutti indipendenti: alcuni, come tm_wday e tm_yday, sono deducibili dagli altri. Il campo finale, tm_isdst, è una flag che indica se attualmente è in corso l'ora legale. La differenza tra gmtime() e localtime() è che la prima ritorna l'orario nella timezone UTC+0 mentre la seconda ritorna nella timezone locale. (**gm** sta per *Greenwich Mean Time*, predecessore di UTC). Ecco un piccolo programma che controlla se è venerdì pomeriggio:

```
#include <stdio.h>
#include <time.h>
main()
{
        time_t t;
        t = time(NULL);
        struct tm *tm;
        tm = localtime(&t);
        if (tm->tm_hour >= 12 && tm->tm_wday == 5)
                printf("E' venerdì pomeriggio!\n");
}
```

Timezone

Probabilmente avete una vaga idea di cosa siano le timezone, anche se i dettagli vi rimangono oscuri. Sono parte del prezzo da pagare per vivere sopra a una sfera rotante. Mezzogiorno a Londra sono le 7 di mattina a New York e via dicendo. Come detto prima, la funzione localtime() converte un orario nella timezone locale. Come sa in che timezone siete? Beh, consulta la variabile d'ambiente TZ:

```
$ export TZ=":CET"
```

imposta la timezone a *Central European Time*. Il valore di questa variabile indica semplicemente un file nella directory ***/usr/share/zoneinfo*** (in questo caso ***/usr/share/zoneinfo/CET***): è questo file che definisce lo scostamento da UTC e le regole di transizione tra orario solare e legale. È un file binario, creato usando il compilatore di timezone (zic), ma può essere esaminato con **zdump**. Modificate il vostro programma per stampare l'ora e il minuto dal tempo destrutturato ritornato da localtime():

```
time_t t;
t = time(NULL);
struct tm *tm;
tm = localtime(&t);
printf("%d:%d\n", tm->tm_hour, tm->tm_min);
```

Ora lanciatelo per timezone diverse:

```
$ export TZ=":GB"
$ ./localtime
21:22
$ TZ=":EST5EDT"
$ ./localtime
16:22
$ TZ=":Europe/Paris"
$ ./localtime
22:22
```

Quindi sono le 21:22 in Inghilterra, cinque ore prima nella timezone "EST5EDT" (New York, ecc.) e un'ora più avanti a Parigi. La timezone "EST" è la stessa di "EST5EDT" eccetto che non ha alcuna informazione sull'ora legale associata. L'esempio "Europe/Paris" mostra un modo alternativo di specificare una timezone. Funziona allo stesso modo, identificando un file in ***/usr/share/zoneinfo***, ma in questo caso il percorso completo sarà ***/usr/share/zoneinfo/Europe/Paris***. Nella maggior parte dei posti, la timezone è scostata da UTC di un tot di ore: Los Angeles è UTC-9, Mosca UTC+3 e via dicendo. Esistono però alcune località in cui questo non è vero. Il Nepal è 5:45 avanti rispetto a UTC. E la Christ Church mantiene la propria timezone che è cinque minuti indietro rispetto a UTC, anche se è difficile che qualche laureando mantenga sincronizzato il proprio orologio.
Se la variabile **TZ** non è impostata, localtime() leggerà le informazioni della timezone da ***/etc/localtime***, che nella maggior parte dei casi nella vostra macchina sarà una semplice copia di ***/usr/share/zoneinfo/Europe/Rome***.

Primavera e autunno

Le timezone non sono l'unico problema nell'ottenere una rappresentazione accettabile del tempo. È stato menzionato »

Aritmetica del tempo

La funzione mktime() può essere usata per convertire un valore "spezzato" in un time_t. In un certo senso è l'inversa di localtime(). La funzione ignora anche i membri tm_wday e tm_yday (comunque deducibili dagli altri). La cosa intelligente di mktime() è che "normalizzerà" i valori, ovvero interpreterà un valore di 68 minuti come 8 minuti oltre all'ora successiva, il 35 gennaio come 4 febbraio e via dicendo. Questo implica anche che potete usarla per fare dell'aritmetica con i valori temporali, per esempio con il codice seguente stamperete l'ora e la data di esattamente una settimana e un'ora dal momento attuale:

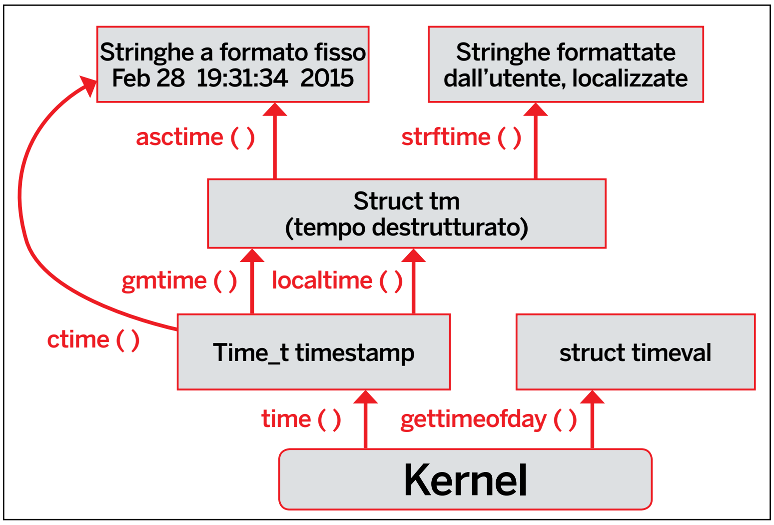
```
time_t t;
struct tm *bdtime;
t = time(NULL);
bdtime = localtime(&t); bdtime->tm_mday += 7; /* Aggiunge una settimana */
bdtime->tm_min += 60; /* e un'ora */
t = mktime(bdtime);
printf("Una settimana più tardi è %s\n", ctime(&t));
```

Usando mktime() potete anche calcolare facilmente il numero di secondi tra due tempi:

```
time_t intervallo;
intervallo = mktime(bdtime1) - mktime(bdtime2);
```

Tip

Fate attenzione a funzioni come **ctime()** che potrebbero ritornare i risultati in uno storage allocato statisticamente. Se pensate di chiamarle più volte, assicuratevi di copiare la stringa ritornata nel vostro storage allocato prima di chiamarle nuovamente; in caso contrario la seconda chiamata potrebbe sovrascrivere il risultato della prima.



› Linux ha diversi modi per rappresentare il tempo, e molte funzioni per convertire tra i vari formati

più volte l'orario legale (DST, *Daylight Saving Time*). Accelerare in primavera e rallentare in autunno, ma le regole per determinare quando succeda sono un po' arbitrarie. Nell'Unione Europea il DST parte dall'ultima domenica di marzo all'ultima domenica di ottobre. Negli USA va dalla seconda domenica di marzo fino alla prima di novembre, a eccezione dell'Arizona (che non ha ora legale). Parti dell'Australia lo adottano anche se, naturalmente è l'opposto, con gli orologi in avanti in ottobre e all'indietro in aprile. Molti altri paesi non osservano l'ora legale. Non è sorprendente che molti dei byte di un tipico file **zoneinfo** servono a specificare, anno per anno, quando comincia e finisce l'ora legale. Fino a ora vi siete concentrati sulla rappresentazione binaria del tempo. Che ne dite di trovare qualcosa da stampare che una persona comune riesca a leggere? Bene, ctime() è un modo facile per farlo, come già detto. Anche questa soluzione ha i suoi limiti: culture diverse hanno nomi diversi per i giorni della settimana e i mesi dell'anno. Gennaio è January in inglese e Janvier in francese, mercoledì è Mittwoch in tedesco.
Per gestire queste e altre differenze culturali, Linux sfrutta i locale. Lo standard Posix indica il locale come "definizione del sottoinsieme di un ambiente utente che dipende dalla lingua e dalle convenzioni culturali". In pratica, un locale definisce convenzioni per visualizzare valute, orari, date e numeri. La posizione geografica che avete specificato durante l'installazione di Linux determinerà il set iniziale dei locale a disposizione, ma potete installarne altri in seguito: in Ubuntu, per esempio, con

```
$ sudo apt-get install language-pack-de
```

installerete il locale per la Germania.
Potete ottenere una lista dei locale a disposizione con il comando $ locale -a. I locale vengono mantenuti in sottodirectory di **/usr/share/locale**.
Ci sono due locali che potete essere sicuri esser presenti, Posix e C, che sono lo stesso e corrispondono al comportamento tradizionale di Unix. Le varie convenzioni culturali che i locale aiutano a definire (date, numeri, valuta, ecc.) sono relativamente indipendenti. Di conseguenza, i locale sono divisi in categorie:

Categoria	Significato
LC_TYPE	Classificazione dei caratteri e regole per la conversione delle maiuscole
LC_COLLATE	Regole di formato alfabetico per un set di caratteri
LC_MONETARY	Regole per visualizzare le valute
LC_NUMERIC	Regole per visualizzare numeri
LC_TIME	Regole per visualizzare orari e date

Un programma può scegliere il locale per una categoria specifica con una chiamata del genere:

```
setlocale(LC_MONETARY, "de_DE.utf8");
```

o potete impostare il locale per tutte le categorie con:

```
setlocale(LC_ALL, "de_DE.utf8");
```

Tuttavia, non è usuale vedere il nome di un locale scritto direttamente in un programma in questo modo. Dopotutto, parte dello scopo dei locale è aiutarvi a scrivere programmi che siano internazionali, ovvero senza assunzioni integrate sulla lingua o sulla cultura. Più di frequente quindi scriverete:

```
setlocale(LC_ALL, "");
```

che prenderà il locale dalla variabile d'ambiente LC_ALL.
In questo modo potete controllare questi aspetti culturali del comportamento di un programma dalla riga di comando semplicemente adattando tale variabile, così:

```
$ export LC_ALL=de_DE.utf8
```

Effettivamente potete controllare separatamente le varie categorie di locale impostando le variabili d'ambiente con nomi corrispondenti, per esempio:

```
$ export LC_MONETARI=fr_FR.utf8
```

imposta solamente la parte di valuta del locale. Notate che se un programma non chiama setlocale(), verrà usato il locale Posix. Ok, ora date un'occhiata alla funzione di conversione dell'ora che sfrutta il locale: strftime(). Questo programma prende un tempo destrutturato e lo formatta in accordo con una stringa di formato specifica. Ecco un esempio:

```
/* Semplice demo di strftime */
#include <stdio.h>
#include <time.h>
#include <locale.h>
int main() {
	time_t t;
	struct tm *bdtime;
	char buf[1000];
	setlocale(LC_ALL, "");
	t = time(NULL);
	bdtime = localtime(&t);
	strftime(buf, 1000, "Sono le %H:%M di %A %d %B
%Y", bdtime);
	puts(buf);
}
```

Avete già visto in precedenza le chiamate a time() e localtime() La chiamata a setlocale() è nuova e la riga più interessante è la chiamata a strftime() . I codici di formato utilizzati qui sono: %H per le ore, %M per i minuti, %A per il nome completo del giorno della settimana, %d per il giorno del mese, %B per il nome completo del mese e %Y per l'anno a quattro cifre. Tuttavia, ci sono molti altri codici di formato e dovreste esplorare la pagina man di strftime. Dal momento che questo programma usa localtime() e strftime() i rispettivi comportamenti sono influenzati dalle variabili TZ e LC_ALL. Per prima cosa provate a lanciarlo con le impostazioni di

default per la vostra posizione:

```
$ unset TZ; unset LC_ALL
$ ./strftime
Sono le 23:07 di Giovedì 03 Settembre 2015
```

Ora passate alla timezone della costa est USA:

```
$ export TZ=":EST5EDT"
$ ./strftime
Sono le 17:08 di Giovedì 03 Settembre 2015
```

Come atteso, l'ora locale è sei ore prima. Avete già visto questo comportamento. Provate cambiando il locale:

```
$ export LC_ALL=fr_FR
$ ./strftime
Sono le 17:09 di Jeudi 03 septembre 2015
```

Qui vedete che il giorno della settimana e il nome del mese sono in francese (e evidenziano che integrare nel codice parole come "Sono le" e "di" non è stato troppo furbo!). Avete uno strano mix in questo caso, con l'orario della costa est ma in francese. Il test finale è quindi riportarlo al CET:

```
$ TZ=":CET"
$ ./strftime
Sono le 23:10 di Jeudi 03 septembre 2015
```

Il problema dell'anno 2038

Molti ricordano il "Millennium bug", causato dalla memorizzazione a due cifre degli anni in computer e database, che portava l'anno 1999 a essere incrementato e diventare 1900, provocando un potenziale disastro e un improvviso boom di richieste di programmatori Cobol. Bene, i sistemi Linux a 32 bit possono rappresentare date da Fri Dec 13 20:45:52 1901 a Tue Jan 19 03:14:07 2038, generando il problema dell'anno 2038. Ovviamente la matematica non è stata fatta a mano, bensì da questo programma:

```
#include <limits.h>
#include <stdio.h>
#include <time.h>
main()
{
        time_t t;
        t = INT_MIN;
        printf("inizio: %s\n", ctime(&t));
        t = INT_MAX;
        printf("fine: %s\n", ctime(&t));
}
```

È possibile che per quella data esisteranno ancora alcuni sistemi Linux integrati a 32 bit, o database che memorizzano un time_t come campo a 32 bit. La maggior parte dei sistemi Linux ancora in circolazione nel 2038 comunque sarà a 64 bit. Se provate a calcolare il tempo più grande rappresentabile da un sistema a 64 bit, modificando INT_MAX in LONG_MAX nel programma sopra, ctime() impazzirà ritornandovi un puntatore nullo. Con un rapido calcolo, stavolta manuale, esce che quei 32 bit in più fanno guadagnare circa 250 miliardi di anni. È rimasto giusto lo spazio per menzionare rapidamente la chiamata times(), che ritorna la quantità di tempo consumata da un processo in esecuzione. Viene ritornata come struttura:

```
struct tms {
        clock_t tms_utime; /* tempo utente */
        clock_t tms_stime; /* tempo di sistema */
        clock_t tms_cutime; /* tempo utente dei child */
        clock_t tms_cstime; /* tempo di sistema dei child */
};
```

Vedete che il tempo è diviso in tempo utente e tempo di sistema, ovvero il tempo speso in spazio utente e il tempo speso all'interno del kernel. La frequenza del clock usata per tali misure è solitamente 100 Hz; se volete sapere qual è realmente nel vostro sistema, sysconf(_SC_CLK_TCK) ve lo dirà. Scaricate questo piccolo programma: **http://linuxformat.com/files/lxf199_sysadmin.c** per una semplice demo. Il programma include molte chiamate a sqrt(), che sapete già essere in spazio utente dal momento che il kernel non è coinvolto nel calcolo della radice quadrata! Ecco un esempio di risultato:

```
$ ./timer
utime = 101
stime = 0
cutime = 0
cstime = 0
```

Come atteso, tutto il tempo viene speso in spazio utente. I tempi dei child sono entrambi zero, chiaramente, dal momento che non ne è stato creato alcuno. Come esperimento finale, commentate il ciclo che chiama sqrt() e decommentate quello che chiama write(). Dal momento che write() è una chiamata di sistema verrà eseguita nel kernel. Con questa modifica, il risultato cambia:

```
$ ./timer
$ ./timer
utime = 19
stime = 147
cutime = 0
cstime = 0
```

Come vi aspettavate, stavolta molto tempo viene speso in spazio kernel. Per ora questo è tutto: il prossimo mese parleremo di processi, concentrandoci su chiamate di sistema fondamentali come fork() ed exec(). Vedrete anche dal punto di vista di un programmatore il più venerabile dei meccanismi di comunicazione interprocesso: la pipe. LXP

Tip

Un PC ha un orologio hardware che mantiene l'ora indipendentemente dall'orologio di sistema governato dal kernel Linux. Il comando **hwclock** offre opzioni per leggere l'orologio hardware e sincronizzarlo con quello di sistema.

Parsing di date e orari

La funzione strptime() può essere usata per convertire la rappresentazione in stringa di una data e ora in una struttura temporale. È per certi versi l'opposto di strftime(). Usa i codici di formato (simili a quelli di strftime) per descrivere il formato di input atteso; chiunque abbia avuto la sfortuna di lavorare con scanf() o fscanf() sa di cosa si parla. Come esempio, ecco un piccolo programma che prende una data nel formato DD/MM/YYYY da riga di comando e vi restituisce il giorno della settimana.

```
/* Calcola il giorno della settimana */
#include <stdio.h>
#include <time.h>
int main(int argc, char *argv[])
{
  char *weekday[] = {
        "domenica", "lunedì", "martedì",
        "mercoledì", "giovedì",
        "venerdì", "sabato"};
  struct tm bdtime;
  strptime(argv[1], "%d/%m/%Y", &bdtime);
  printf("Il giorno è %s\n", weekday[bdtime.
tm_ wday]);
}
```

Usando questo programma potete sapere in che giorno della settimana cadrà Natale quest'anno:

```
$ ./day-of-week 25/12/2015
Il giorno è venerdì
```

Ora lo sapete. Naturalmente dovrete preparare i regali sotto l'albero già da giovedì sera.

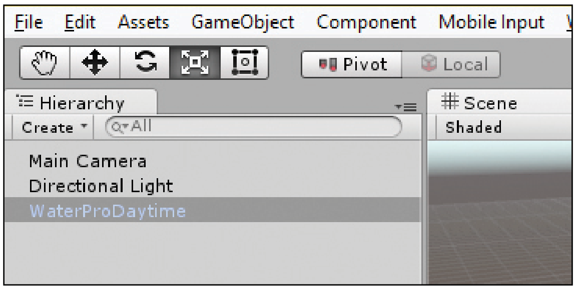
Creare un gioco 3D per Linux

Una guida passo passo per creare il vostro primo gioco con l'SDK del più popolare game engine gratuito

Per anni Linux è stata una piattaforma priva di giochi, ma quei giorni neri sono ormai finiti. Lo sviluppo di giochi, d'altra parte, è un po' indietro; anche Unity, il popolare kit di sviluppo che ha esportato centinaia di giochi per Linux non ha ancora il supporto nativo. Quindi, anche se con un groppo al cuore, eccovi **Unity**; supporta Linux come engine ma non come piattaforma di sviluppo. Se volete vedere più giochi in Linux dovete usare un SDK basato su Windows. Grazie al cielo **Unity 5** gira in Linux con **Wine**. In Ubuntu o Linux Mint il processo è lineare, ma dovrete fare attenzione che se non imposterete una flag specifica a 0 l'installazione fallirà con un errore fatale. Lanciate questo comando come root:

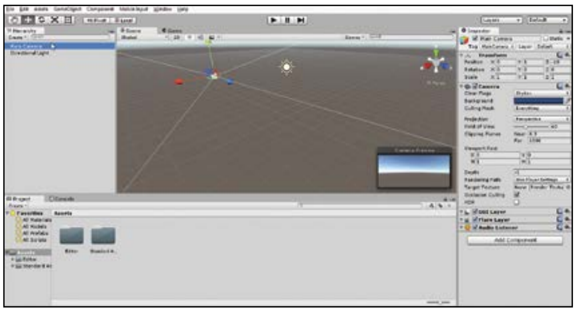
```
$ echo 0 | sudo tee /proc/sys/kernel/yama/ptrace_scope
```

Per recuperare Unity 5 e lanciarlo installate **PlayOnLinux** tramite il vostro centro software. Al primo avvio PlayOnLinux vi potrebbe chiedere alcune cose, tipo l'installazione di font extra, che potete accettare per continuare. Potete scaricare l'installer Unity per Windows da **http://unity3d.com/unity/download**. Unity viene installato con uno script PlayOnLinux che potete salvare in GitHub da **http://bit.ly/LXPUnity**. In PlayOnLinux scegliete **Tools** -> **Run a local script** e scegliete il file **Unity.pol** appena scaricato. Seguite lo script di installazione, dal momento che non è ufficiale dovrete prima accettare la responsabilità relativa. Lo script si occupa di installare Wine e diversi componenti Windows che occuperanno qualche gigabyte di spazio, dopo la finestra che vi chiederà di accettare (o riparare) l'installazione. Vi verrà quindi chiesto di individuare l'eseguibile dell'installer Windows e procederà all'installazione di Unity 5. Una tazza di tè più tardi e, dopo l'installazione, avrete bisogno di un account Unity creabile dal sito Web. Ora potete lanciare Unity dall'icona desktop o PlayOnLinux, scegliete e create un account Unity gratuito, entrate e siete pronti al via.



› ***Unity* ha una barra degli strumenti basilare per le trasformazioni, da sinistra a destra: visualizza, trasforma, ruota, scala, seleziona**

Aprite Unity e scegliete **New Project**. Chiamatelo **IslandRun** e nello stesso schermo cliccate sul pulsante **Asset Packages**. Spuntate le checkbox per **Cameras**, **Characters**, **Environment** e **Vehicles** e quindi scegliete **Done**, quindi **Create Project**. La libreria degli asset del vostro gioco ci metterà un po' a caricarsi, processo accompagnato dalla barra di progressione. Una volta nell'editor Unity vedrete una schermata simile a quella di qualsiasi software di modellazione 3D. Sopra avrete la scena principale all'interno della quale piazzare i vostri oggetti 3D o 2D. Potete crearli utilizzando strumenti quali **Gimp** e **Bender** e importarli, ma in questo tutorial verranno usati degli asset prefabbricati dalle librerie che avete appena caricato.

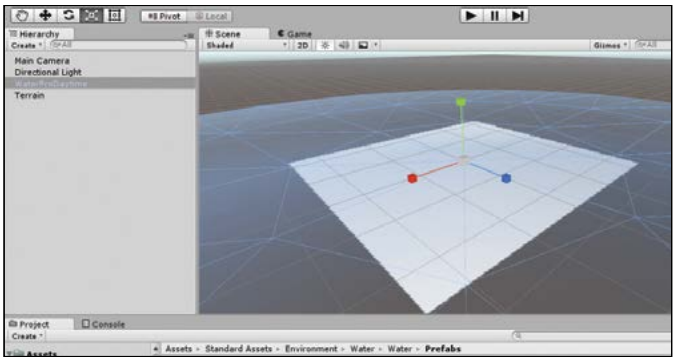


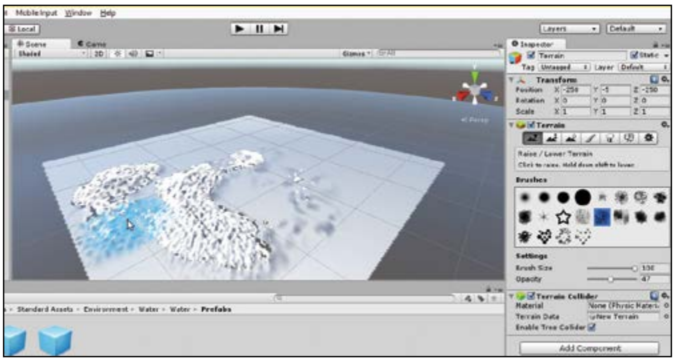
› **Se non avete lo stesso layout dello schermo della figura, andate in Window -> Layouts -> Default**

Impostare la scena

Pensate alla vista Scena come al set di un film in cui assemblate i componenti del gioco prima del ciak. Se cliccate sulla scheda con l'etichetta **Game**, la finestra cambierà nella vista gioco. Continuando l'analogia del set di un film, la vista vi mostrerà solamente quello che l'audience vedrà attraverso le videocamere piazzate sulla scena. Cliccate nuovamente sulla scheda **Scene**. Vicino alle finestre sulla sinistra c'è una lista di tutto quello che è attualmente nella scena. Pensatelo come l'inventario dell'aiuto regista. Se cliccate sulla camera nella lista verrà evidenziata la camera corrispondente nella scena. Scrivere **cam** nella casella di ricerca isolerà qualsiasi cosa nella lista contenente quelle lettere, facendone un utile modo per trovare cose, come una camera, nella vostra scena. »

Creare il terreno e l'isola



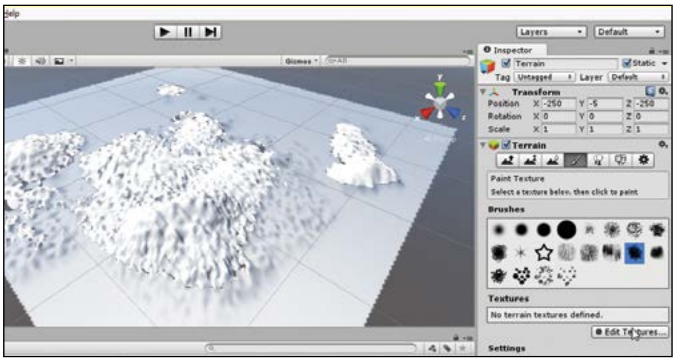


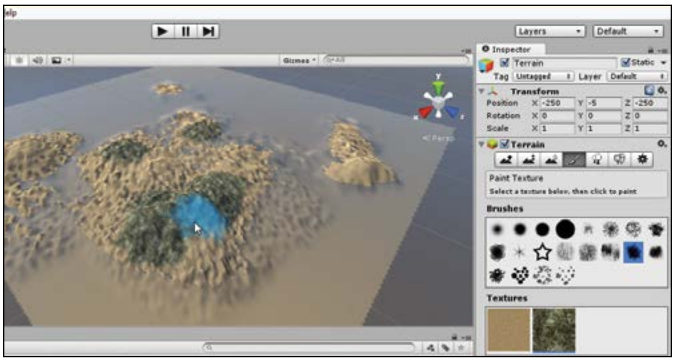
1 Creare e centrare il terreno

Unity contiene strumenti per la modifica del terreno che userete ora per creare un'isola. Andate in **Game Object** -> **3d Object** -> **Terrain**. Nell'**Inspector** inserite **X -250 Y -5** e **Z -250**. Questo centrerà il terreno nel mezzo dell'acqua già, dal momento che il terreno per default è un quadrato grande 500. Il **-5** nell'asse Y permette all'acqua di stare sopra al terreno.

2 Dipingere l'isola

A questo punto aumentate l'acqua in modo che si allunghi sopra il terreno. Selezionatela e inserite **X 1000** e **Z 1000**. Ora solleverete varie parti del terreno per creare l'isola. Selezionato il terreno, cliccate il pulsante **Raise/Lower Terrain**. Selezionate uno dei pennelli grezzi e incrementate la dimensione a 100 e l'opacità a 50. Dipingete sul terreno la forma generale della vostra isola.





3 Aggiungere dettagli

Dipingete sul terreno la forma generale della vostra isola. Riducete la dimensione e l'opacità del pennello, quindi ripetete per aggiungere dettagli medi. Infine, usate **Smooth Height** se avete un terreno troppo accidentato. Usate il pulsante per lo scroll per fare zoom in e out e il pulsante centrale del mouse per spostarvi nella vista. Cliccate **Paint Texture** e selezionate **Edit Textures** e **Add Texture**.

4 Applicare e dipingere le texture

Cliccate **Select** sul quadrato a sinistra etichettato **Albedo** (RGB). Scorrete in giù per trovare **SandAlbedo** e selezionatelo. Scegliete **Add**. Notate che la sabbia sarà applicata a tutto il terreno. Ora aggiungete un'altra texture allo stesso modo, scegliendo tra **Cliff** o **Rocky**. Selezionatela e usate i pennelli come prima, dipingendo sulla sommità del terreno sabbioso la vostra visione dell'isola.

Esportare il gioco per Linux

Il vostro gioco dovrebbe essere pronto per essere impacchettato e provato sulla vostra macchina Linux. Salvate la Scena e il progetto. Ora andate su **File** -> **Build Settings**. Scegliete **PC Mac & Linux Standalone**. Scegliete **Linux** come **Target Platform** e se volete x86 per macchine a 32 bit o x86_64 per macchine a 64 bit. Cliccate **Add Current** per aggiungere la scena. Cliccate **Player Settings**. Inserite il nome della vostra azienda e tutti i dettagli che volete, come uno splash screen. Cliccate su **Build** e inserite il nome del file, quindi **Save**.
Se trasferite il file in Linux ricordate di non lanciarlo direttamente da un memory stick perché probabilmente non funzionerà. Copiatelo sul desktop e assicuratevi che il file binario abbia permessi di esecuzione. Fate click con il tasto destro sul file, scegliete **Proprietà** e nella scheda **Permessi** spuntate la voce per permetterne l'esecuzione. Ben fatto, avete creato il vostro primo gioco 3D in Linux con Unity 5!

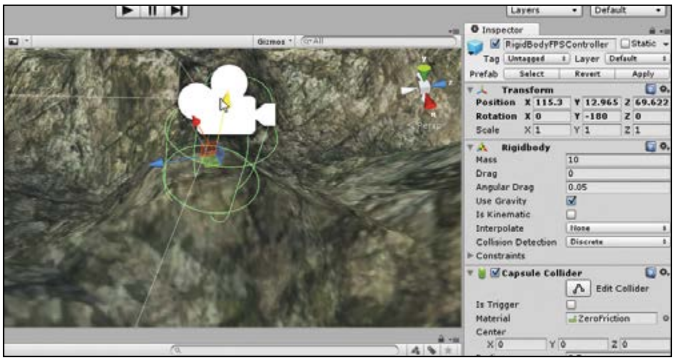


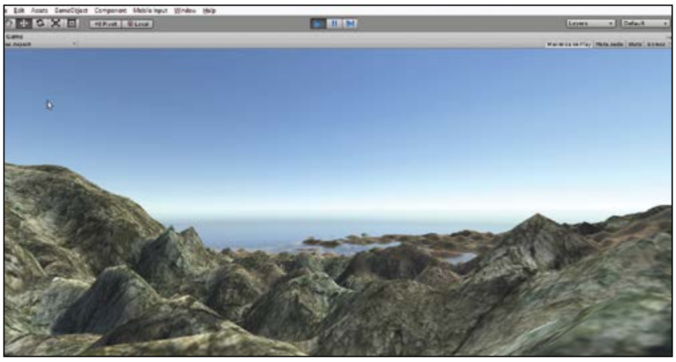
› Le possibilità di esportazione per il vostro gioco sono molte ma scegliete saggiamente: questa è una rivista Linux...

Tip

Unity 5 contiene un sole, una camera e un cielo in ogni scena. Provate a pensare a una scena di Unity come a uno spazio nel mondo reale: se non aveste cielo, luce del sole o occhi per guardare, il vostro mondo sarebbe oscuro.

Aggiungere il giocatore



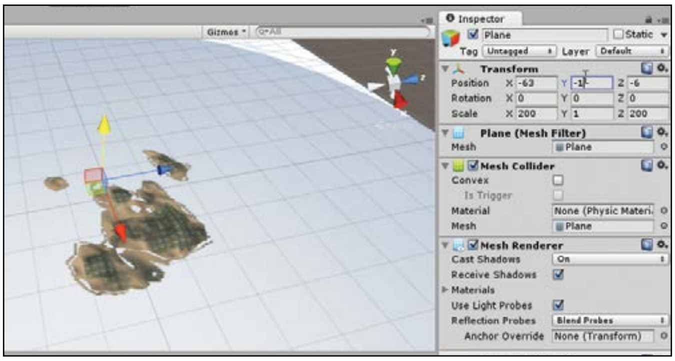


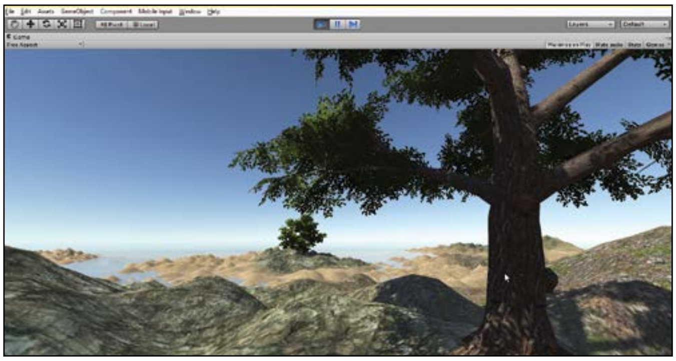
1 Aggiungere il personaggio

Una scena senza giocatore non è molto interattiva, quindi aggiungete un Player Controller così da camminare per la vostra scena. Nella scheda **Project** andate in **Assets** -> **Standard Assets** -> **Characters** -> **FirstPersonCharacter**-> **Prefabs** e trascinate il controllo **RigidBody** sul vostro terreno. Zoomate e con il pulsante **Move** selezionato trascinate l'asse giallo in su finché il box verde non è a livello del terreno.

2 Camminare per l'isola

Ora potrebbe essere una buona opportunità per fare un piccolo test drive della vostra isola! Cliccate sulla scheda **Game** e quindi su **Play**. Se vedete che il vostro giocatore cade attraverso il terreno, tornate indietro alla Scena e alzate ancora un po' il giocatore. Premendo **Play** dovreste poter camminare per l'isola e vederne le bellezze.



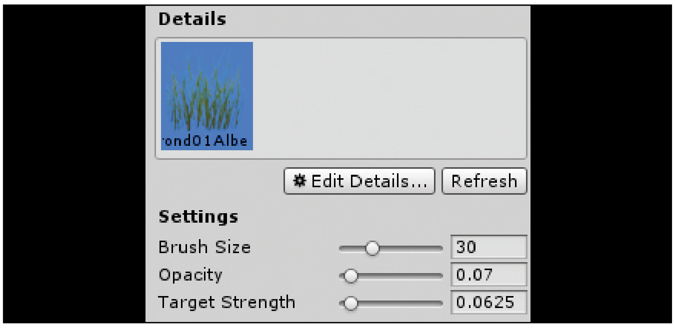


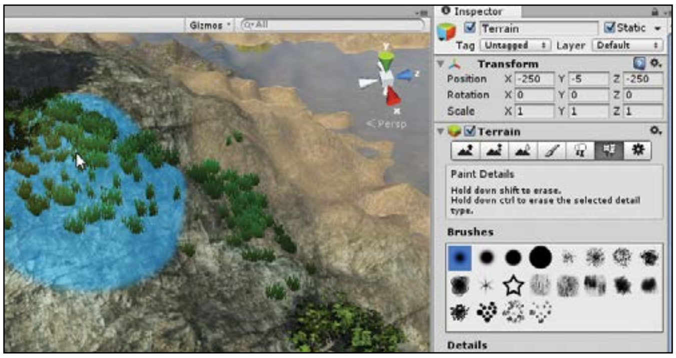
3 Misure anti-annegamento

Avrete notato che il giocatore può camminare nell'acqua e scomparirvi. Aggiungete un altro 3D Object, stavolta selezionando **Plane**. Scalatelo di 200 in X e Z, quindi impostate la **Transform** di Y a -1, portandolo appena sotto l'acqua. Spuntate **Mesh Collider** ma non Mesh Renderer. Avete appena creato un piano invisibile appena sotto il pelo dell'acqua con un set **Collider** per impedire a qualsiasi cosa di andare sotto acqua.

4 Aggiungere gli alberi

Ora aggiungete alcuni particolari d'ambiente alla scena creando degli adorabili alberi. Nella sezione **Terrain** nell'Explorer, scegliete **Place Trees** e cliccate **Edit Trees** -> **Add Tree**. Cliccate il piccolo cerchio nella finestra di **Add Tree**. Scegliete **Broadleaf Desktop** e cliccate **Add**. Impostate la dimensione del pennello a 20 e la densità dell'albero a 50. Dipingete sul terreno dove volete disporre i vostri alberi.





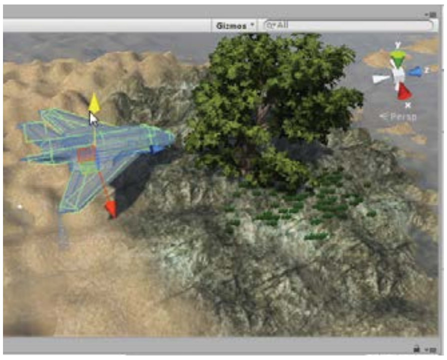
5 Creare l'erba

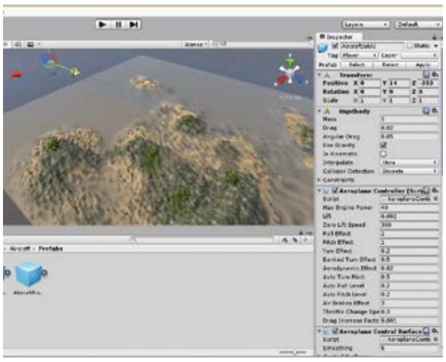
Andate al pulsante seguente chiamato **Paint Details**. Cliccate **Edit Details** e **Add Grass Texture**. Cliccate il cerchio e fate doppio click su una delle texture **GrassFrond**. Scegliete **Healthy Color** e rendetelo un po' meno luminoso. Fate lo stesso con **Dry Color**. Cliccate **Add**. Modificate la dimensione del pennello, opacità e **Target Strength** rispettivamente a 30, 0.07 e 0.07.

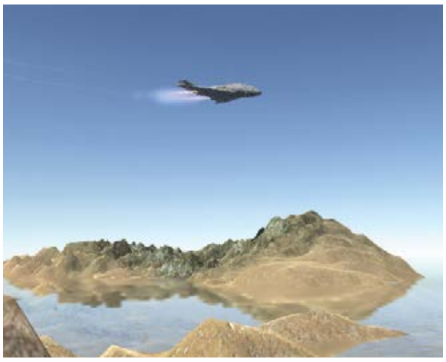
6 Mettere erba dettagliata o cespugli

Usate i pennelli come prima e dipingete l'erba dove volete vederla. Provate il gioco e vedete come risulta, quindi tornate alla scena e aggiungetene o toglietene tenendo premuto **Shift** finché dipingete. Una cosa da tenere a mente è che l'erba viene mostrata solo quando siete vicini, quindi dovrete fare zoom durante la pittura.

Aggiungere un aereo







1 Aggiungere l'aereo

Ora aggiungerete un aereo nemico che caccerà il giocatore. Questo aereo seguirà la posizione del giocatore, così da costringerlo a muoversi sempre invece di ammirare la vista. Andate nuovamente in **Standard Assets** e cercate i prefab per Aircraft. Trascinate **AircraftJectAI** o **AircraftPropellerAI** nella vostra scheda **Hierarchy**. Fateci doppio click per centrare la vista sull'aereo.

2 Impostare l'AI

Dovrete usare il pulsante **Translate** (in alto a sinistra, secondo pulsante) per muovere l'aereo. Volete muoverlo sopra l'acqua e lontano dal mare. Nell'**Inspector** scorrete fino a **Aeroplane AI Control** e scegliete il cerchio per **Target**, quindi scegliete **RigidBodyFPSController**. Tornando all'**Inspector** dovrete smarcare **Waypoint Progress Tracker**.

3 Creare una barriera al volo

A questo punto provate il gioco. L'aereo punterà a voi e probabilmente si schianterà al suolo. Per sistemare questo problema, scegliete il **Plane invisibile** e cliccate con il tasto destro su **Duplicate**. Spostate la copia a Y 35 o quello che funziona per il vostro terreno. Abilitate temporaneamente il **Mesh renderer** per vedere l'aereo e assicurarvi che non sbatta con alcuna elevazione del vostro terreno.

Sulla estrema destra dello schermo trovate l'**Inspector**, che mostrerà i dettagli della camera selezionata. Pulite il box di ricerca e selezionate la **Directional Light** nell'elenco **Hierarchy**. Vedrete i dettagli cambiare a quelli più specifici per la vostra illuminazione. Non c'è nulla da vedere al momento, quindi cominciate con l'acqua, così da aggiungere qualcosa di interessante da vedere e dei confini naturali al vostro mondo. Sotto la finestra della scena c'è la lista del progetto, contenente tutti gli asset del gioco caricati nel progetto. Scegliete **Standard Assets** -> **Environment** -> **Water** -> **Water** -> **Prefabs**, quindi trascinate l'elemento **WaterProDaytime** nell'elenco **Hierarchy** e apparirà nella scena. Sopra, a sinistra della finestra della scena, avete cinque pulsanti. Selezionate il quarto, cliccate nella piccola box al centro del prefab dell'acqua e tenete premuto il pulsante sinistro del mouse finché muovete in su. Questo scalerà l'acqua. Il prefab Water che avete inserito nella scena è un disco piatto con un materiale già applicato (vedete il box). Cliccate sul pulsante **Play** in centro allo schermo e vedete cos'altro ha da offrirvi. Avete notato che premendo **Play**, Unity esegue uno script che anima l'acqua? Quello che avete fino a ora è una camera statica puntata all'orizzonte di una scena che ha una luce, un cielo che simula luce diffusa, e un piano piatto con materiale acqua applicato mentre uno script di animazione ne anima le onde. Andate su **File** -> **Save Scene** e scrivete **IslandRun_Scene** premendo **Save**. Seguite le guide di queste pagine per aggiungere terreno, giocatori e nemici al vostro nuovo gioco! LXP

Come funzionano i materiali

Le grafiche dei giochi sono migliorate moltissimo nel corso degli anni ma il formato di base è rimasto lo stesso. Gli oggetti sono fatti di mesh, a loro volta fatte di poligoni, in genere triangoli o quadrilateri. Pensatelo come la struttura di un gazebo con sopra il classico tendone. Tutto quello che ottenete è una forma 3D ma nessun dettaglio o colore a cui assegnare i materiali applicati. I materiali consistono di texture, chiamata **Albedo** in Unity 5. Contiene le info base del colore. In aggiunta ci sono degli extra come una mappa normale che definisce dettagli come rilievi, crepe, botte e graffi. Un materiale può avere anche una mappa di occlusione ambientale, che indica a Unity dove sono le aree di ombra costante o una mappa speculare che controlla quale parte del materiale risulta lucida. Una mappa di altezza viene usata, come quella del terreno qui, dove l'elevazione è controllata dal livello di grigio in una immagine in scala di grigi. Leggete l'ottima introduzione su **http://bit.ly/Unity5StandardShader** oppure **http://bit.ly/Unity5PhysicallyBasedShading**.



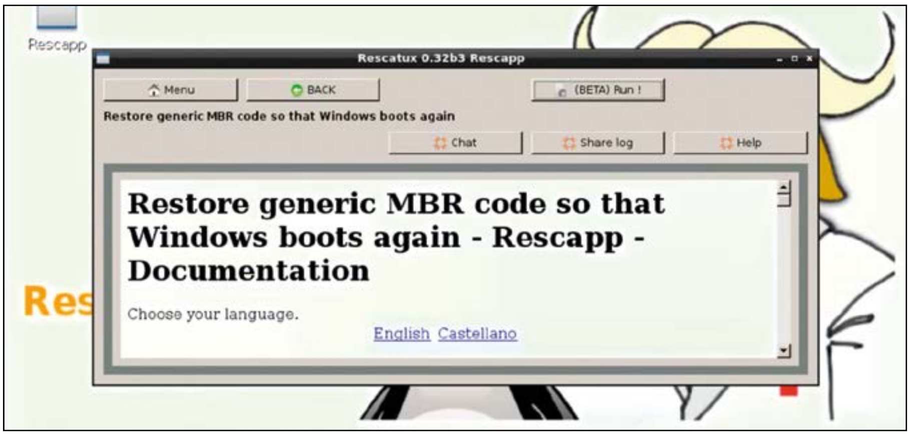
› ***Unity 5*** **contiene una scena demo per aiutarvi a capire i materiali e l'illuminazione**

Domande & Risposte

Avete qualche domanda sull'Open Source? Scriveteci a **mailserver@linuxpro.it** per ottenere una risposta

1 Avvio bloccato

Ho acquistato un Acer TMP253/M con Windows 7 Professional 64 bit. Ho già utilizzato Linux in passato e l'ho anche consigliato a molti miei amici a cui faccio assistenza. Tuttavia, è la prima volta che mi trovo a installarlo in presenza di un BIOS recente. Volevo montare Ubuntu 14.10, mantenendo comunque Windows sulla stessa partizione. Ho disabilitato FastBoot, ma a questo punto non viene più rilevata alcuna presenza di Windows. Ho provato a utilizzare il comando *sudo gdisk dev/sda*, ma non sono riuscito a risolvere. Il PC, peraltro, mi chiede di inserire un disco di avvio.



› ***Rescatux* è una distro live ottimizzata per il ripristino del sistema. Al suo interno sono disponibili numerosi strumenti, tra cui *Testdisk***

Utilizzando il comando **gdisk**, molto probabilmente è stata eliminata la tabella delle partizioni. Ciò ha quindi impedito a Windows di avviarsi. Tuttavia, a meno che tu non abbia scritto sul disco, i filesystem sono ancora presenti. Manca solo la tabella che indica la loro posizione. Per fortuna c'è uno strumento che scansiona hard disk e partizioni, per poi ricreare la tabella. Stiamo parlando di **Testdisk**. Devi eseguirlo da un CD live che lo contiene come **Rescatux**. Scorri fino a giungere nella parte inferiore della finestra **Rescapp** e troverai la voce **Testdisk** nella sezione **Expert Tools**. Una volta lanciato, seleziona l'opzione per creare un file di registro che può essere sempre utile nel caso qualcosa non funzioni. A questo punto, scegli il disco, di solito **/dev/sda**. **/dev/sr0**, invece, si riferisce quasi sempre all'unità DVD. Seleziona il tipo di tabella per le partizioni che dovrebbe essere **EFI GPT**. A questo punto, potrai vedere la struttura della partizione attuale che probabilmente sarà vuota (visto che l'hai precedentemente cancellata). Premi la funzione **Quick Search**. Nonostante il nome, questa opzione potrebbe richiedere un po' di tempo per essere eseguita. Se il computer non risponde, non c'è niente di cui preoccuparsi: è del tutto normale.
Una volta completata la scansione, Testdisk mostrerà cosa ha trovato. Premi **Invio** per passare alla schermata successiva. Se ciò che è emerso dalla scansione ti soddisfa, seleziona **Write** per salvare la tabella delle partizioni. In alternativa, scegli **Deeper Search** per una scansione ancora più approfondita. Anche in questo caso, i tempi di esecuzione della procedura possono essere particolarmente lunghi. Infine, premi **Quit** e riavvia il computer. Non è detto che il problema si risolva, in quanto il danno potrebbe essere più esteso. Quella descritta è comunque una delle procedure

Riflettori su...

Si può aggiornare il kernel senza ricompilarlo?

Certamente sì, la procedura per aggiornare il kernel non richiede una nuova ricompilazione. Basta semplicemente utilizzare il comando

```
# apt-cache search linux-image
```

che fornisce una lista di kernel già compilati e pronti a essere scaricati. Dopo aver scelto quello che preferite (attenzione: occorre optare per la release adatta in base all'architettura del processore), basta installarlo come un qualsiasi altro pacchetto:

```
# apt-get install linux-image...
```

Prima di aggiornare il vostro kernel, valutate sempre i benefici di cui potreste godere. Non è detto che un kernel più recente sia per forza migliore di quello che già state usando.
Se avete Debian, potete provare **Vanilla** che trovate alla pagina **www.kernel.org**.

Terminale e superutente

Nelle distro Linux ci sono diversi modi per applicare un comando a una data operazione. La maggior parte delle volte, infatti, potete procedere sia utilizzando degli strumenti grafici sia la potenza della riga di comando. In quest'ultimo caso, avrete a disposizione strumenti molto più flessibili e pratici da usare, ma soprattutto validi per tutte le distribuzioni. Prendendo come esempio i comandi per la configurazione di sistema, avrete sicuramente notato come nel momento in cui tentate di eseguirne uno, vi venga richiesto l'accesso come *superuser* o *root*. In Ubuntu e nelle sue derivate, per esempio, questo è identificato con il comando **sudo**, cui poi deve seguire l'inserimento della password. Altre distro come Fedora, invece, utilizzano **su**, che danno accesso all'utente root fino a quando non si inserisce il comando **logout**.

consigliate in questi casi. A tal proposito, prendiamo spunto per ricordarvi sempre di eseguire un backup incrementale dei vostri documenti più importanti. Questo è solo un esempio di come un disco fisso possa essere vulnerabile.

2 Dubbio su codice

Ho letto il vostro articolo "Metti subito il turbo a Linux!" pubblicato su Linux Pro 146. Quando tento di eseguire le istruzioni riportate nel seguente codice da voi fornito (pag. 12), compare questo errore:

```
sudo sed --in-place 's/NoDisplay=true/
Nodisplay=false/g' *.desktop
```

```
sed:-e expression#1,char35: unknown
option to `s
```

Purtroppo non ho abbastanza esperienza per trovare una soluzione. Ho iniziato a usare Linux da poco e quindi non so capire se c'è un errore di battitura, oppure si tratta di un problema relativo al mio sistema. Potreste cortesemente spiegarmi cos'è successo?

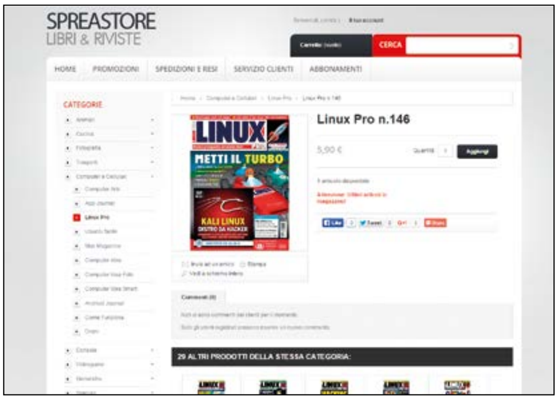
La stringa riportata funziona correttamente. Il suo ruolo è prendere uno script e applicarlo a un input. In questo caso, lo script è

```
s/NoDisplay=true/
Nodisplay=false/g
```

La **s** all'inizio indica a **sed** di eseguire una ricerca con relativa sostituzione. La parte compresa tra i primi due backslash è da sostituire, mentre quella che segue contiene il testo sostitutivo. Il backslash finale è poi richiesto per la corretta esecuzione del comando. La funzione **g**, invece, sostituisce solo la prima istanza su ogni riga. Normalmente, **sed** legge da un pipe o da una serie di file specificati in linea di comando, quindi riporta il proprio input adatto al reindirizzamento a un file o un pipe di un altro programma. L'opzione **--in-place** indica la necessità di modificare il file in loco (in realtà, scrive l'output in un file temporaneo per poi sostituire l'originale). Non sapendo quale distro stai usando e la relativa versione, è difficile sapere se si tratta di un problema relativo al tuo sistema. Ciò non toglie che le versioni più obsolete di **sed** non accettano file di input in combinazione con il parametro **--in-place**. In tal caso, dovrai eseguire il comando singolarmente per ogni file. Tuttavia, è possibile risolvere con un breve loop:

```
for i in *.desktop; do
  sed -i 's/NoDisplay=true/
NoDisplay=false/g' "$i"
  done
```

Questo esegue il comando una volta per ogni file corrispondente. Le virgolette tra il marcatore sono lì nel caso i nomi dei file contengano degli spazi. Si può eseguire



› Sul numero 146 di Linux Pro c'è l'articolo "Metti subito il turbo a Linux!". Si può acquistare da http://bit.ly/1KcEhwT

una procedura simile anche con **find**:

```
find -name '*.desktop' -exec sed 's/
NoDisplay=true/NoDisplay=false/g' "{}" \;
```

Troverai così tutti i file che corrispondono al modello inserito. A questo punto, basterà eseguire il comando **exec** per ognuno, sostituendo le due parentesi graffe con il nome del file.

3 Mancato avvio

Per liberarmi dal terribile Windows Vista, ho formattato il disco e installato Linux Mint 17. Purtroppo, però, l'avvio non è andato a buon fine. Utilizzando un altro PC con Mint 13, lanciando poi md5sum, ho scoperto che il download è andato a buon fine. Ho quindi applicato *sudo fdisk -l distro.iso* e scoperto che esiste un boot track. Questo, nonostante le dimensioni, è vuoto.

```
Device Boot
Start      End     Blocks   Id  System
/media/20150112_155405/linuxmint-17.1-
cinnamon-64bit.iso1   *        0      3026591
1513296    0  Empty
/media/20150112_155405/linuxmint-17.1-
cinnamon-64bit.iso2       2958600     2963143
2272   ef  EFI (FAT-12/16/32)
```

Nel tuo caso, l'immagine ISO è la soluzione »

Riferimento rapido

Completamento automatico avanzato

Ci sono una serie di motivi per cui molti rifuggono l'uso del terminale. In primo luogo, è necessario ricordarsi i comandi. In seconda battuta, è obbligatorio scriverli correttamente perché funzionino. Fino a quando siete in presenza di istruzioni corte e semplici, non ci sono problemi, ma se avete necessità di usare istruzioni più lunghe e articolate, allora le cose si complicano. Ecco quindi che ci viene in aiuto il completamento automatico avanzato. Si tratta di una funzione che permette di completare l'inserimento di un'istruzione o di un percorso attraverso l'uso del tasto **TAB** nella shell **Bash**. Se provate a inserire un comando qualsiasi come **apt-get install gimp**, premendo una sola volta il tasto **TAB**, il sistema completerà l'istruzione nel caso non ci siano altri modi per definirla. Se invece premete due volte **TAB**, compare una lista di svariate possibilità da usare per completare la stringa. Nel caso di Gimp, per esempio, troverete **gimp-cbmplugs**, **gimp-data**, **gimp-data-extras** e molti altri. Se la lista è molto lunga, come nel caso in cui proviate a premere due volte **TAB** dopo un comando tipo **apt-get install kde**, il sistema riporterà il numero dei record trovati e chiederà l'autorizzazione a mostrarli.

migliore da sfruttare. Le varie cautele sull'uso di **fdisk** relative a una ISO si applicano solo se vuoi utilizzare una chiave USB. Il fatto che il boot track sia vuoto non deve allarmarti. Infatti, indica solo che la partizione non dispone di un filesystem ID. Come puoi vedere dai valori di inizio e fine, è ricco di informazioni. Hai provato ad avviare il disco dalla macchina in cui hai installato Mint 13? Se il checksum della ISO è corretto, le cause più probabili di questo problema possono essere due: una configurazione scorretta del BIOS o del firmware EFI che impedisce di eseguire il boot da DVD. In alternativa, è il lettore di supporti ottici a essere difettoso e quindi incapace di leggere i dischi inseriti. Queste due cause possono essere valutate semplicemente usando un altro PC. C'è anche la possibilità che il DVD con la ISO di Mint sia stato masterizzato malamente. Ciò può essere dovuto o a un errore nel processo, oppure all'uso di un supporto di scarsa qualità. Questo soprattutto se si tratta di un disco riscrivibile che spesso presenta parecchi problemi di compatibilità anche con i lettori più flessibili. In tal caso, ti consigliamo di provare di nuovo a masterizzare il supporto, magari usando un DVD-R. Per sicurezza, riduci al minimo la velocità di masterizzazione portandola a 2x. Questo può evitare l'insorgenza di potenziali errori di scrittura. Tra le altre cose, è possibile verificare rapidamente il funzionamento di un'immagine ISO tramite l'emulatore **Qemu**. Per usarlo, installa il pacchetto **qemu-kvm**. Utilizza quindi uno dei seguenti comandi, tenendo presente che il primo è più veloce. Lancia poi il secondo solo se il primo non funziona.

```
kvm -boot d -cdrom linuxmint-17.1-cinnamon-64bit.iso
qemu-system-x86_64 -boot d -cdrom linuxmint-17.1-cinnamon-64bit.iso
```

Per l'occasione puoi usare anche VirtualBox.



› **Con *dynDNS.it* è possibile registrare e gestire un proprio DNS dinamico gratuitamente. Per chi ha bisogno di ulteriori funzioni, sono disponibili piani a pagamento**

La soluzione del mese

Connessione al server

Sto cercando di eseguire un server sulla LAN che mi dia accesso a *RStudio*, così da poterlo usare dall'iPad quando sono lontano da casa. Allo stato attuale, il tutto funziona perfettamente fino a quando mi trovo all'interno della rete domestica. Quando però sono fuori e uso Internet, non c'è niente da fare. Il collegamento al server non riesce. Sto utilizzando il server di RStudio su Linux Mint 14.1. Quando provo ad accedere al server dalla LAN con http://192.168.0.108:8787, la connessione è immediata. Se invece tento di raggiungerlo dal Web con http://73.213.144.65:8787, mi viene restituito l'errore *404 page not found*. Il mio IP è corretto e l'ho rilevato da uno dei tanti siti abilitati a questo scopo. Potete aiutarmi a capire dove sbaglio e cosa dovrei fare? Ho anche impostato un indirizzo IP statico (192.168.0.108), nonché eseguito un port forwarding sulla porta 8787 e disabilitato UPnP.

Questo genere di problema, di solito, viene causato da un'errata configurazione del port forwarding. Se però hai già valutato questa possibilità e controllato che tutto sia a posto, allora valutiamo altre strade. In primo luogo, ricordati che se utilizzi un IP dinamico, il tuo ISP può cambiarlo senza nessun preavviso. Potresti quindi connetterti a un indirizzo che nel momento in cui viene interpellato non è più valido. In seconda battuta, bisogna poi considerare che determinati provider tendono a bloccare alcune porte per motivi di sicurezza. Contatta quindi il tuo fornitore di servizi e richiedi informazioni riguardo a questa eventualità. Nel caso ci siano dei blocchi attivi, è necessario utilizzare un altro computer al di fuori della Rete. Una delle migliori soluzioni che si possono adottare è l'uso di un tunneling SSH inverso. L'unico requisito è avere una distro Linux con **sshd** (anche se a onor del vero è possibile eseguire la procedura anche con altri sistemi operativi). Il file di configurazione **/etc/ssh/sshd-config** deve poi contenere la seguente riga:

```
GatewayPorts yes
```

Il PC, inoltre, deve avere un indirizzo o un nome di dominio statico (è possibile utilizzare un servizio che fornisce DNS dinamici. Tra i tanti, possiamo consigliarti **dynDNS.it**. Sono disponibili account gratuiti e a pagamento in base alle funzioni di cui vuoi usufruire). Apri quindi il tunnel sul server, eseguendo:

```
ssh -R 8000:localhost:8787 utente@altrocomputer
```

Visto che questo comando viene eseguito sul server, non sarà influenzato da eventuali blocchi su porte in entrata. Adesso, quando ti connetti a **http://altrocomputer:8000**, tutto il traffico verrà inoltrato alla porta 8787 del server. Non hai quindi bisogno di utilizzare altre porte, né di eseguire alcun port forwarding. Questa procedura richiede un computer esterno e l'uso di un VPS (*Virtual Private Server*) che tuttavia può essere disponibile per pochi euro al mese.



› ***EasyBCD*** **per Windows non è gratuito, ma ha un costo di 29,95 dollari (26 euro)**

4 Bootloader di Windows

Dopo aver provato Linux, ho deciso di installare Ubuntu sul mio secondo disco fisso. Tuttavia, quest'ultimo dispone già di una partizione NTFS che voglio usare per mantenere il boot loader di Windows. Potreste indicarmi come districarmi nella procedura d'installazione di Ubuntu per consentirmi di usare anche Windows?

Se il tuo PC dispone di un sistema UEFI, non ci sono problemi. Infatti, quest'ultimo funziona anche da bootloader, quindi avrai solo bisogno di caricare un boot manager che di solito è disponibile premendo il tasto **Func** o **Esc** all'avvio del computer. Questo è possibile, poiché UEFI consente di utilizzare più boot manager sullo stesso disco. Se però hai un sistema basato su BIOS/MBR, le cose si complicano un po'. Le precedenti versioni di Ubuntu mettevano a disposizione un'immagine alternativa per l'installazione. Questa permetteva di installare il bootloader di Ubuntu nella partizione di root. Se vuoi procedere in questo senso ti consigliamo quindi di usare **Lubuntu**. In tal modo, il bootloader di Windows rimarrà intatto e avrai comunque la possibilità di gestire diverse opzioni di avvio. Per farlo, è necessario aggiungere una voce al menu di avvio di Windows. Il sistema più semplice è utilizzare **Windows EasyBCD** (**http://neosmart.net/EasyBCD**). Avvia quindi il sistema Microsoft, installa il programma ed eseguilo. Fai click su **Add Entry** e seleziona **Grub 2** dalle opzioni. Seleziona **Write MBR**. In tal modo, dovresti avere a disposizione un menu di avvio che include anche l'opzione Grub per Ubuntu. In caso poi si voglia eliminare Ubuntu e ristabilire il bootloader Microsoft, basta riutilizzare **EasyBCD**. LXP

Soluzione rapida

La quantità di distro a disposizione degli amanti del mondo Linux è davvero enorme. Non passa giorno in cui non vengano rilasciate nuove versioni. Per chi si avvicina per la prima volta al Pinguino, può non essere facile scegliere quella più adatta per le proprie competenze e per l'uso che se ne intende fare. Un primo consiglio che diamo, è di dare un'occhiata a **Distrowatch**, **www.distrowatch.com**. In seconda battuta, potete fare un semplice test che vi dirà quale distro è più adatta a voi. Lo trovate alla pagina **www.linux.it/scegli**. In linea generale, comunque, le migliori distribuzioni per i principianti sono Ubuntu e Linux Mint.





FAQ: le domande più frequenti

Virtual Private Network

› È un nome particolarmente fantasioso, ma di cosa si tratta in realtà?
Una **Virtual Private Network** o VPN è una rete all'interno di un'altra rete.

› Sono impressionato, ma non ho ancora ben chiaro il concetto.
Le VPN vengono principalmente utilizzate per trasferire dati su una rete privata che viaggia all'interno di una pubblica come Internet. Il collegamento attraversa un tunnel criptato. La sicurezza della rete pubblica non può essere garantita, così come non si ha nessun controllo sull'accesso all'hardware intermedio, ma la cifratura dei dati li rende inaccessibili.

› Perché dovrei aver bisogno di usare una VPN?
Immagina di avere una rete nel tuo ufficio coperta da un firewall e di volerti collegare da casa. Una VPN stabilisce una connessione tra il computer casalingo e la rete aziendale crittografando i dati che transitano. In tal modo, pur utilizzando Internet, al suo interno stai sfruttando un'altra rete criptata. È come se passassi in un tunnel a cui non ha accesso nessuno.

› C'è bisogno di hardware particolare?
Avrai bisogno di un server VPN in esecuzione sulla rete, quindi di una configurazione adeguata del firewall. Infine c'è bisogno di un software client VPN che viene eseguito in locale.

› Sembra costoso
In realtà no. Molti router anche economici dispongono del supporto per VPN. Attenzione solo ai dispositivi marchiati *VPN pass-though*. Questi, pur supportando l'uso di una VPN, non sono in grado di gestire una connessione di questo tipo e quindi ci sarà sempre bisogno di un client.

› In definitiva, di quale software ho bisogno per usare una VPN?
Dovrai dotarti di un client e un kernel opportunatamente configurato. La maggior parte delle distro sfruttano già kernel adatti allo scopo. Il software necessario, tuttavia, dipende molto dalla variante di VPN che viene usata all'altra estremità del collegamento. Di solito **OpenVPN** (**http://openvpn.net**) e **Openswan** (**www.openswan.org**) sono perfetti per la maggior parte delle situazioni.



L’eco dei LUG

I Lug

I LUG rappresentano da sempre il punto di riferimento per chiunque voglia conoscere GNU/Linux. Ogni mese dedicheremo loro questo spazio per la comunicazione di nuovi progetti e appuntamenti. Se hai qualcosa da segnalarci scrivi a **ecodeilug@linuxpro.it**

ABRUZZO
AnxaLUG - Lanciano
www.anxalug.org
Il Pinguino - Teramo
Non disponibile
MarsicaLUG - Marsica
www.marsicalug.it
OpenLUG - L’Aquila
Non disponibile
Pescara LUG
www.pescaralug.org
Pineto LUG
www.pinetolug.org
Pollinux LUG - Pollutri
Non disponibile
SSVLUG - San Salvo, Vasto, Termoli
www.ssvlug.org
SulmonaLUG
http://sulmonalug.it
TeateLUG - Chieti
Non disponibile
TeLUG - Teramo
www.telug.it
User Group Valle Roveto
http://linuxvalley-os4.blogspot.com/

BASILICATA
Basilicata LUG - Potenza e Matera
www.baslug.org

CALABRIA
3BYLug - Trebisacce
www.3bylug.tk
Bogomips - Bisignano
www.blug.it
CastroLUG
http://castrolug.altervista.org
Cosenza Hack Laboratory
http://hacklab.cosenzainrete.it/
CSLUG - Cosenza
http://cslug.linux.it
CzLug
Non disponibile
HackLab Catanzaro
http://hacklab.cz
Piana LUG - Piana di Gioia Tauro
Non disponibile
Reggio Calabria LUG
http://rclug.linux.it
Revolutionary Mind
www.revolutionarymind.org
SpixLug - Spezzano Albanese
Non disponibile

CAMPANIA
AFR@Linux LUG
www.afralinux.netsons.org
Afralug - Afragola
www.afralug.com
CasertaLUG
www.casertaglug.org
Hackaserta 81100
www.81100.eu.org
HackMeetNaples Napoli HackLab
www1.autistici.org/hmn
IGLUG - Napoli e provincia
www.iglug.org
IRLUG - Irpinia
www.irlug.it
LUG-Ischia
www.lug-ischia.org
NALUG - Napoli
www.nalug.net
Neapolis Hacklab
www.officina99.org/hacklab.html
Padulug - Paduli (BN)
http://linux.paduli.com
SCALUG - Scafati (SA)
http://xoomer.alice.it/scalug/
Tuxway.org - Provincia di Napoli
www.tuxway.org
ValuLug - Vallo Linux User Group
www.valug.it
XALUG - Salerno
http://xalug.tuxlab.org

EMILIA ROMAGNA
ALFLUG - Alfonsine
www.alflug.it
Borgotaro LUG - Val Taro
http://btlug.it/
ConoscereLinux - Modena
www.conoscerelinux.it
ERLUG
http://erlug.linux.it
Ferrara LUG
www.ferrara.linux.it
FoLUG - Forlì
http://folug.linux.it
ImoLUG - Imola
www.imolug.org
LUGPiacenza
www.lugpiacenza.org
PANLUG - Vignola
Non disponibile
PLUG - Parma
http://parma.linux.it
RavennaLUG
www.ravennalug.org
RELug - Reggio Emilia e provincia
http://relug.linux.it
RiminiLug
www.riminilug.it
S.P.R.I.Te
http://sprite.csr.unibo.it
UIELinux - Valle del Rubicone
www.uielinux.org

FRIULI VENEZIA GIULIA
CarniaLUG - Tolmezzo
www.carnialug.net
GOLUG - Gorizia
www.golug.it
IGLU - Udine
http://iglu.cc.uniud.it
LUG Pordenone
www.pnlug.it
LugTrieste
http://trieste.linux.it
LUG [A] [L] [P] - Aquileia
www.alproject.org

LAZIO
CiLUG - Frosinone
www.cilug.org
CLUG - Cassino
http://cassino.linux.it/
GioveLUG - Terracina
www.giovelug.org
La Sapienza LUG
www.lslug.org
Latina LUG
www.llg.it
LUG Privernum Volsca - Priverno (LT)
www.pvlug.org
LUGRoma
www.lugroma.org
LUGRoma 3
www.lugroma3.org
TorLUG - Università Tor Vergata - Roma
http://lug.uniroma2.it/
V.I.S.C.O.S.A. - Ciampino
www.viscosa.org

LIGURIA
Genuense Lug - Genova e d’intorni
http://genova.linux.it
GinLug - Genova Sampierdarena
www.sennaweb.org
Govonis GNU/LUG - Provincia di Savona
www.govonis.org
SLIMP - Software Libero Imperia
http://slimp.it/
TLug-TSL - Tigullio Ligure
http://tlug.linux.it/

LOMBARDIA
BGLug - Bergamo e provincia
www.bglug.it
BGLug Valle Seriana - Valle Seriana
http://bglugvs.web3king.com/
BrigX - Monza e Brianza
http://brigx.it
GL-Como - Como
www.gl-como.it
GLUX - Lecco e provincia
www.lecco.linux.it
GULLP - Gruppo Utenti Linux Lonate Pozzolo
www.gullp.it
IspraLUG - Ispra
http://ispralug.eu/
LIFO - Varese
www.lifolab.org
LIFOS - Cinisello Balsamo
www.lifos.org
Linux Var - Varese
www.linuxvar.it
LoLug - Lodi e provincia
www.lolug.org
Lug Bocconi - Milano
www.lug-bocconi.org
LugBS - Brescia e provincia
http://lugbs.linux.it/
Lug Castegnato - Castegnato

L'eco dei LUG

www.kenparker.eu/LugCastegnato
LugCR - Cremona e provincia
www.lugcr.it
Lug Crema - Crema
http://filibusta.crema.unimi.it/
LUGDucale - Vigevano
www.lugducale.it
LugMan - Mantova e provincia
www.lugman.org
LugOB - Cologne e ovest bresciano
www.lugob.org
MoBLUG - Monza e Brianza
www.bubblesfactory.it
OpenLabs - Milano
www.openlabs.it
POuL - Milano
www.poul.org
TiLug - Pavia
http://pavia.linux.it
VigLug - Vignate, Milano Est - Adda Martesana
www.viglug.org

MARCHE

Ascolinux LUG/FSUG Ascoli
http://marche.linux.it/ascoli/
CameLUG - Camerino
www.camelug.it
CMlug
www.cmlug.org
Egloo
www.egloo.org
FanoLUG
www.fanolug.org
Fermo LUG
www.linuxfm.org/fermolug/
GLM - Macerata
www.gruppolinuxmc.it/start/index.php
LUG Ancona
www.egloo.org
LUG Marche
http://marche.linux.it
PDP Free Software User Group
http://pdp.linux.it
SenaLug - Senigallia
www.lug.senigallia.biz

MOLISE

Campobasso LUG
http://cb.linux.it/
FrenterLUG - Larino
Non disponibile
SmaLUG - San Martino
www.smalug.org

PIEMONTE

ABC Lug - Alba/Bra/Carmagnola
http://abc.linux.it/
AlLug - Alessandria e provincia
www.allug.it
BiLUG - Provincia di Biella
http://www.bilug.it
FASoLi - Alessandria e provincia
http://softwarelibero.al.it/
Gallug - Galliate
www.gallug.it
GlugTO - Torino e provincia
www.torino.linux.it
IvLug - Ivrea Linux User Group
www.ivlug.it
Linox Novara
www.linoxnovara.org
SLIP - Pinerolo
http://pinerolo.linux.it/
ValSusinux - Val Susa e Val Sangone
www.valsusinux.it

PUGLIA

BriLUG - Brindisi
www.brilug.it
CapitanLUG - Capitanata
www.capitanlug.it
LATLUG - Latiano Linux User Group
www.latlug.org
LUGargano
www.lugargano.it
LUGBari - Bari e provincia
www.lugbari.org
MurgiaLug - Santeramo in Colle
www.open-pc.eu/index.php/murgialug/
SaLUG! - Salento
http://salug.it
Talug - Taranto
www.talug.it

SARDEGNA

GNUraghe - Oristano
www.gnuraghe.org
GULCh - Cagliari
www.gulch.it
PLUGS - Sassari
www.plugs.it
GULMh - Macomer (NU)
www.gulmh.org

SICILIA

CefaLug - Cefalù
http://cefalug.linux.it
cLUG - Caltanissetta
www.clug.it
EnnaLUG
www.ennalug.org
FreakNet MediaLab - Catania
www.freaknet.org
Leonforte LUG
http://leonforte.linux.it
LUG Catania
www.catania.linux.it
LUGSR - Siracusa
www.siracusa.linux.it
MELUG - Messina
Non disponibile
Norp LUG - Noto, Pachino, Rosolini
Non disponibile
PALUG - Palermo
http://palermo.linux.it
RgLUG - Ragusa e provincia
http://ragusa.linux.it
VPLUG Linux Planet - Provincia Caltanisetta
www.vplug.it
SputniX - Palermo
www.sputnix.it

TOSCANA

ACROS - Versilia, Lucca, Massa Carrara
www.lug-acros.org
Elbalinux
Non disponibile
ElsaGLUG - Val d'Elsa
www.elsaglug.org
FLUG - Firenze
www.firenze.linux.it
GOLEM - Empoli, Valdelsa
http://golem.linux.it
GroLUG - Grosseto
www.grolug.org
G.U.L.LI - Livorno
www.livorno.linux.it
GulP! Piombino
http://gulp.perlmonk.org
GULP Pisa
www.gulp.linux.it
GuruAtWork - Grosseto e provincia
www.guruatwork.com
IPIOS - Bibbiena e valle del Casentino
www.ipios.org
Lucca LUG
http://luccalug.it
L.U.G.A.R - Arezzo
Non disponibile
PLUG - Prato e provincia
www.prato.linux.it
PtLug - Pistoia e provincia
www.ptlug.org
SLUG - Siena e provincia
www.siena.linux.it

TRENTINO ALTO ADIGE

LinuxTrent - Trento
http://linuxtrent.it
LugBz - Bolzano
www.lugbz.org

UMBRIA

OrvietoLUG
www.orvietolug.it
LUG Perugia
www.perugiagnulug.org
TerniLUG
www.ternignulug.org

VALLE D'AOSTA

SLAG - Aosta
www.slag.it

VENETO

0421ug - Provincia di Venezia
www.0421ug.org
BLUG - Belluno
http://belluno.linux.it
Faber Libertatis - Padova
http://faberlibertatis.org
GrappaLUG - Bassano del Grappa
http://grappalug.homelinux.net/
ILC - Informatica Libera Cittadellese - FSUG
http://ilc.pd.it
LegnagoLUG
Non disponibile
Linux Ludus - Villafranca (VR)
www.linuxludus.it
LugAnegA
www.luganega.org
LUGSF - San Fidenzio
Non disponibile
LUG Vicenza
www.vicenza.linux.it
LugVR - Verona
www.verona.linux.it
MontelLUG - Montebelluna
www.montellug.it
FSUG Padova
www.fsugpadova.org
RoLUG - Rovigo
http://rovigo.linux.it
TVLUG - Treviso
www.tvlug.it
VELug - Venezia
www.velug.it
AViLUG Schio
http://www.avilug.it/doku.php

NAZIONALI

FSUGitalia
www.fsugitalia.org
Gentoo Channel Italia
www.gechi.it
MajaGLUG
www.majaglug.net
SkyLUG
http://tech.groups.yahoo.com/group/skylug/

Guida DVD

Ogni mese Linux Pro vi offre i programmi e le distribuzioni più recenti su DVD

Ogni volta che troverete questo simbolo in un articolo, vorrà dire che i file citati si trovano nel DVD allegato alla rivista.

Distro Desktop

Netrunner 16

Netrunner 16 Ozymandias è una distro disponibile in due versioni: standard basata su Kubuntu e Rolling Release progettata invece su Manjaro. In questo caso ci siamo concentrati sulla prima, mettendola a disposizione nel DVD di questo numero. L'approccio semplice e intuitivo che presenta fin dal primo avvio permette di gestire al meglio le tante applicazioni disponibili. Infatti, è principalmente studiata per fornire software multimediali e una vasta schiera di programmi utili all'uso quotidiano.

Le principali novità

La distro è disponibile sia in versione a 32 bit sia a 64 e usa il kernel 3.19.0 ~16. Se scaricata, ha un peso di circa 1,8 GB. Netrunner 16 sfrutta il desktop Plasma 5.3, nonché applicazioni KDE 15.04. Questa versione di Plasma non può ancora considerarsi perfetta, tuttavia sta dimostrando di essere una notevole evoluzione rispetto a KDE4. Oltre a migliorare la qualità dell'approccio e l'esperienza d'uso, ha potenziato anche la stabilità generale. Adesso, infatti, è possibile interfacciarsi con i widget presenti sulla scrivania in modo ancora più veloce e semplificato. Possiamo muoverli, ridimensionarli o configurarli facendo click con il tasto sinistro del mouse sopra di essi. I widget sfruttano Carbon, uno stile derivato da Oxygen con temi predefiniti per GTK2 e GTK3, nonché per Qt5 e Qt4. I requisiti minimi per l'utilizzo di Netrunner 16 sono tutt'altro che pretenziosi. È sufficiente un processore 1,6 GHz Intel Atom, 1 GB di RAM, un disco fisso con almeno 10 GB di spazio libero, una scheda grafica Intel GMA 945 con memoria video di almeno 128 MB.
Se vogliamo provare Netrunner in VirtualBox, gli sviluppatori consigliano di impostare un quantitativo di RAM pari o superiore ai 1.024 MB. Se la distro con architettura a 64 bit dovesse restituire un crash in fase di installazione, il team di Netrunner consiglia di usare **Image Writer** per masterizzare la ISO.
In più, è necessario disattivare la funzione **Secure Boot** dal UEFI.

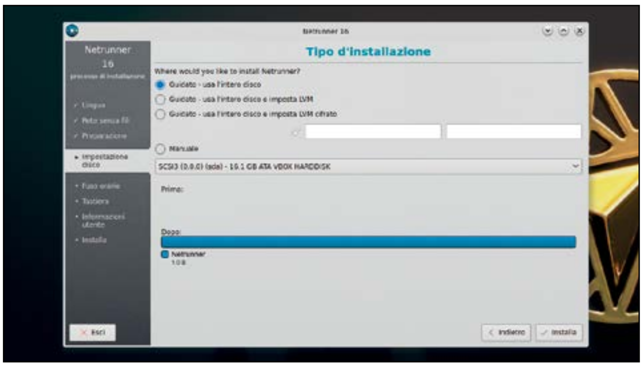
Programmi per tutti

Le applicazioni disponibili sono tante. Nell'immediato, quindi, non sentirete il bisogno di accedere al software manager per scaricarne altre. Per quanto riguarda la navigazione, troviamo **Firefox 37.0**, **Thunderbird 31.6** che vengono peraltro mantenuti costantemente aggiornati grazie al repo **Plasmazilla** (**ppa:plasmazilla/releases**). Ci sono poi **Skype**, **Pidgin 2.10.9**, **Transmission-Qt** e **Marble**, un'ottima applicazione per la mappatura che permette di tracciare percorsi e interfacciarsi con i dati GPS. Per gli amanti della musica, troviamo **Gmusicbrowser**, **Audacious** e **Radiotray**. Il primo è un juke-box Open Source, utile quando avete a che fare con collezioni musicali particolarmente vaste. Radiotray, invece, è un'applicazione minimale e molto leggera che permette di sintonizzarsi sulle principali radio che trasmettono tramite Internet. Per quanto riguarda la parte multimediale, potete sfruttare **KDEnlive**, **Handbrake**, **Vokoscreen 2.1** e **VLC Media Player 2.2**. Per chi non lo conoscesse, Vokoscreen è un pratico software per registrare tutto ciò che avviene sullo schermo. È molto utile se siamo soliti registrare tutorial da pubblicare poi su YouTube. Infine, per chi vuole divertirsi, sono presenti **Steam**, **Playonlinux** e vari altri giochi free tipo Mahjonng, FrozenBlubble e GLTron. Il supporto in caso di bisogno è estremamente vario e completo. Sul sito ufficiale disponibile alla pagina **www.netrunner.com**, troviamo un forum, più una serie di tutorial. Questi analizzano nel dettaglio la procedura d'installazione e in più forniscono una serie di video per l'utilizzo delle applicazioni più interessanti.

Cosa c'è nel DVD

- Debian Live 8.0.1
- DNSCypt 1.6.0
- Guake 0.8.0
- Megaglesx 3.11.1
- Minilens
- OpenIndiana 151a8
- Plank 0.9.1
- QmapShack 1.3.0
- SecurePoint 10
- Spl 0.6.4.2
- Vdrift 2012-07-22
- Tor Browser 5.02

Cosa ci offre Netrunner 16



1 Installazione intuitiva

L'installer vi guida in una procedura guidata estremamente semplice e veloce. Dopo aver scelto la lingua e verificato i requisiti hardware minimi, potrete procedere con il partizionamento del disco. A questo proposito, sono disponibili le procedure manuale e guidata.



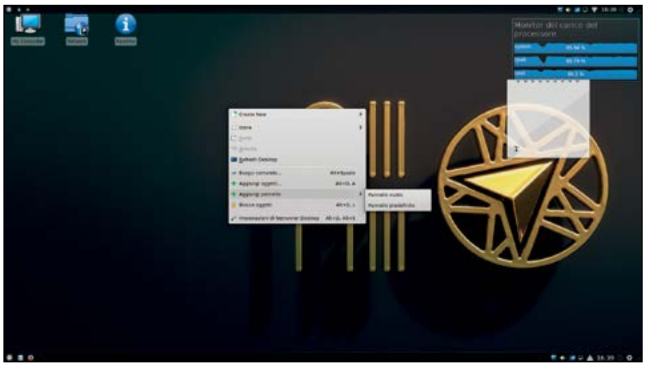
2 Desktop ordinato

Il desktop KDE 5 mette subito a proprio agio. I principali programmi sono accessibili tramite il pulsante di avvio presente a sinistra nella barra degli strumenti. A destra, invece, avete le funzioni di accesso rapido che per esempio consentono di avviare al volo il terminale.



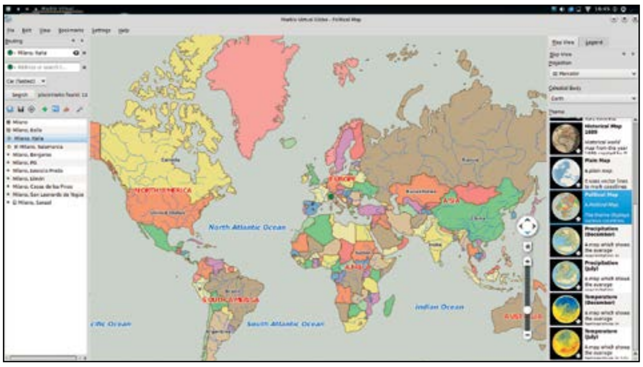
3 Mondo widget

I widget sono una parte integrante del desktop. Potete aggiungerli facendo click con il tasto destro del mouse sulla scrivania e selezionando la voce **Aggiungi oggetto**. Si apre così una barra sulla sinistra che li elenca tutti. Ce ne sono una gran quantità e per tutte le esigenze.



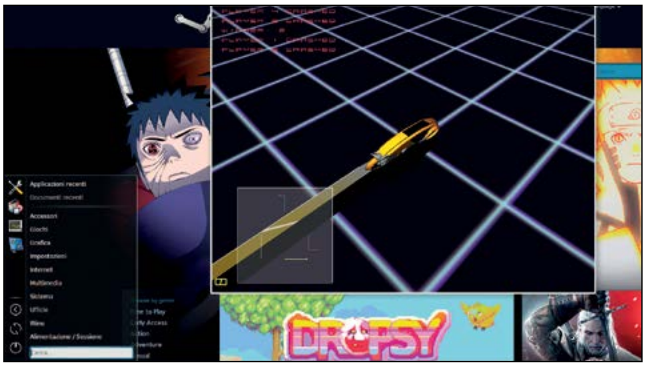
4 Doppia barra degli strumenti

Facendo click con il tasto destro del mouse sulla scrivania, potete selezionare la voce **Aggiungi pannello**. Nella parte superiore compare quindi una nuova barra degli strumenti che va ad aggiungersi a quella inferiore. È poi possibile scegliere se implementarne una vuota o già dotata dei vari programmi predefiniti.



5 Il mondo in uno schermo

Marble è un'applicazione per la gestione cartografica. Agendo sul menu di destra, è possibile cambiare le impostazioni visive del planisfero. In questo esempio, abbiamo usato la visualizzazione politica, ma sono disponibili molte altre voci tra cui quella relativa al tempo atmosferico e alla temperatura.



6 Firewall configurabile

Per gli amanti dei videogiochi, **Steam** è già installato nel sistema e pronto all'uso. Non mancano altri classici come **GLtron** che vi ripropone la classica corsa con le moto vista nel film **Tron Legacy**. C'è poi **PlayOnLinux** che permette di giocare con una grande varietà di titoli. LXP

In edicola il 12 novembre

NEL PROSSIMO NUMERO

Hai un argomento da proporci?
Scrivi a prossimamente@linux.it



IL MEGLIO DEL 2015

Abbiamo messo alla prova tutte le migliori distro di quest'anno per trovare quali dovete assolutamente installare e quali sono più adatte alle vostre esigenze!

LINUX PRO

Mensile, prezzo di copertina 5,90 €
www.linuxpro.it

Direttore responsabile: *Luca Sprea*

Traduzione e Localizzazionea cura di:
Ventidodici di Andrea Orchesi
redazione@linuxpro.it

Sprea S.p.A.
Socio Unico - direzione e coordinamento di Sprea Holding S.p.A.

Presidente: Luca Sprea
Consigliere delegato: Mario Sprea

Coordinamento:
Gabriella Re (Foreign Rights) international@sprea.it, Alberta Rivolta (PA), Ambra Palermi (Segreteria Editoriale), Francesca Sigismondi (Ufficio Legale), Tiziana Rosato (acquisti e produzione), Emanuela Mapelli (Pianificazione Pubblicitaria)

Amministrazione: Anna Nese (CFO), Erika Colombo (controller), Irene Citino, Sara Palestra
amministrazione@sprea.it

Servizio qualità edicolanti e DL: Sonia Lancellotti, Andrea Palermi
distribuzione@sprea.it

PUBBLICITA' E MARKETING:
Alessandra Cappellacci- alessandracappellacci@sprea.it
Tel.0292432275 - Cell.3204670523

Sede Legale: - via Torino, 51 20063 Cernusco Sul Naviglio (Mi) - Italia
PI 12770820152- Iscrizione camera Commercio 00746350149

ABBONAMENTI E ARRETRATI
Servizio qualità abbonamenti e arretrati: Desirée Conti

Abbonamenti: si sottoscrivono on-line su www.linuxpro.it/abbonamenti
Mail: abbonamenti@linuxpro.it
Fax: 02 700 53 76 72
Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)

Arretrati: si acquistano on-line su www.linuxpro.it/arretrati
Mail: arretrati@linuxpro.it
Fax: 02 700 53 76 72 - Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)
Il prezzo dell'abbonamento è calcolato in modo etico perché sia un servizio utile e non in concorrenza con la distribuzione in edicola.

www.myabb.it a cura di Aktia srl
Contenuti su licenza: "Linux Format"- Future Publishing Limited PLC., Bath UK

Registrazione testata: Linux Pro, pubblicazione mensile registrata al Tribunale di Milano il 08.02.2003 con il numero 74.

Distributore per l'Italia e per l'estero:
Press-Di Distribuzione stampa e multimedia s.r.l. - 20134 Milano
ISSN: 1722-6163

Stampa: Arti Grafiche Boccia S.p.A.- Salerno


PROBLEMI CON
MUFFA E CONDENSA?
SMUFFOLO
LA TUA CASA RESPIRA
www.smuffolo.it
FACILE DA INSTALLARE
LIBERA LA CASA DA ALLERGENI
FA RESPIRARE GLI AMBIENTI
ECONOMICO ECOLOGICO
ordini@smuffolo.it
02 80886782

